博士論文

Early Recognition of Emerging and Disappearing Entities in Microblogs for Entity-based Social Listening

（エンティティに基づくソーシャルリスニングのためのマイクロブログにおける新規および消失エンティティの早期認識）

48-187401

赤﨑　智

指導教員　豊田　正史

大学院　情報理工学系研究科　電子情報学専攻

東京大学

This dissertation is submitted for the degree of

Doctor of Philosophy

December 2021

Acknowledgements

はじめに、指導教官である豊田正史先生に感謝いたします。 自由気ままに研究生活を送る自分に対し小言の一つも言わず見守っていただき、普段の研究指導だけでなく、種々の申請書類の手続きや研究室の環境整備など沢山の物事について配慮してくださりました。 研究においては、勉強不足の私にソーシャルメディア分野の知識を授けてくれるだけでなく、論文や申請書で重要な研究のストーリーの書き方についても根気強く指導していただき、自分がいかにその部分を疎かにしていたか学ぶことができました。 博士課程をここまで歩んでこれたのは間違いなく豊田先生のお力添えあってのことです。

次に、同じ研究室のスタッフであり、実質的に自分のもう一人の指導教官である吉永直樹先生に感謝いたします。 自然言語処理分野の研究をやるにあたり、知識と経験に裏打ちされた吉永先生の指導なしでは思いつかなかった手法やできなかった実験が沢山あります。 論文や申請書の執筆の際も私の拙い作文を何度も何度も添削してもらい、投稿にこぎつけるまでしっかりと面倒を見ていただきました。 また、研究指導だけでなく日常生活においてもカジュアルに食事や飲み会に誘っていただき、親身に色々な相談に乗っていただきました。 吉永先生もまた、私の博士課程において欠いてはならない恩人です。

Yahoo! JAPAN 研究所の鍛治伸裕さんには、博士課程の間のインターンのメンターとして面倒を見ていただきました。 企業の中での研究というものがどういうものなのか、論文の書き方や良い会議に通すコツなど、様々なことを教えていただきました。 前期課程にインターンで指導していただいた研究が初めてのトップ会議に通ったおかげで研究活動に勢いがつき、後の論文や申請書の採択に繋がったのは間違いないです。 本当にありがとうございました。

また、ここまで何不自由ない環境で伸び伸びと研究ができたのは、喜連川優先生が研究室の環境とスタッフの方々を裏側から支えてくださったおかげです。大変忙しく滅多に会うこともなかったですが、数少ないアドバイザミーティングの際は貴重なアドバイスを頂けました。「自分の研究を俯瞰的に見た時にどういう立ち位置にあるのか、何十年と時間が経った時にどう貢献しているのか、振り返って満足できるのか」というお言葉を今後も胸の片隅に留めながら研究を続けていきたいと思っています。ありがとうございました。

本博士論文の主査の鶴岡慶雅先生、副査の山肩洋子先生、森純一郎先生には、お忙しいにも関わらず審査を引き受けて頂きいくつもの貴重な意見や助言をいただきました。 先生方の助力のおかげで、本論文を自分の納得の行く出来に仕上げることができました。感謝いたします。

その他の元も含めた研究室の教員や研究員の方々、合田和生先生、横山大作先生、伊藤正彦先生、小宮山純平先生、梅本和俊先生、石渡祥之佑さん、佐藤翔悦さんには、ミーティングでの助言、論文や申請書へのコメント等を頂きました。これだけの人達から手助けしてもらえる環境にいたことは本当に貴重だと思います。ありがとうございました。

研究室の歴代の秘書の方々である井﨑葉子さん、周佐亜樹さん、池田鈴子さん、越水さちよさんにも感謝申し上げます。 科研費の処理や出張費の精算など面倒な手続きを引き受けてくださり、大変助かりました。 秘書さん方の協力がなければ、事務作業に忙殺されて研究どころではなかったと思います。

在学生の方々にもたくさんお世話になりました。特に共著で一緒に研究を行った大葉大輔さん、論文の英語の添削をしていただいたジョシュア・ベンジャミン・ターナーさんに感謝いたします。彼らのお力添えのおかげで出せた研究成果もありました。

最後に、奔放な私の学生生活をこれまで見守り支えてくださった両親と祖母に、特別な感謝を捧げます。これまで本当に長い間、色々と心配をかけました。

書き割りの様なチープで出来損ないの私の夢の世界、それでも、博士課程で過ごせた時間は...間違いなく素晴らしき日々でした。

Abstract

With the spread of smartphones and the appearance of microblogs such as Twitter, it has become possible for all kinds of people to send posts anytime, anywhere, and all kinds of information, from real-world things to users' personal affairs, can now be sent out in the form of text. The knowledge posted in these natural languages has various values. It is helpful to society in multiple ways, such as for people to track things they are interested in and for mass media and local governments to understand incidents or disasters. For this reason, gathering such information through microblogs, called social listening, has become an essential factor of microblog applications such as social trend analysis, marketing research, and entity recommendation.

Social listening requires knowledge of entities that ceaselessly emerge and disappear in the world. Monitoring the emerging entities, which are newly born in the world, such as new products, works, companies, and events, leads to understanding the real-time trends in the real world. In contrast, the disappearing entities that are ceasing to exist in the world, such as famous people who have passed away, stores that are closing or going bankrupt, products and events that are being discontinued, also play an important role in decision making about entities.

Although large knowledge bases such as Freebase and Wikipedia can serve as a reference for understanding these entities, they are not comprehensive because they rely on manual labor to register entities and describe specific knowledge about them. In addition, since only notable entities are registered in those knowledge bases, knowledge about long-tail but useful entities such as rising stars, local events, and stores is often overlooked. Therefore, without relying on specific resources like knowledge bases, it is necessary to discover emerging entities and disappearing entities in microblogs as soon as they (are scheduled to) appear and disappear, regardless of their notability or frequency. At this point, for subsequent social listening applications, since it is not enough to perform entity discovery only, we have to estimate and attach attribute information such as an entity type from

microblog posts. This is especially essential when targeting emerging entities since we cannot extract that knowledge using existing knowledge bases.

Based on this background, we address the following three issues:

Discovery of emerging entities

New events, works, and products that constantly appear in the real world are essential for various applications such as marketing research and entity recommendation due to the novelty of their information. However, such entities cannot be treated as knowledge since they do not exist in existing corpora or knowledge bases. As a first step to overcome this situation, we discover emerging entities from microblogs as soon as they appear. In this case, the challenge is how to detect only emerging entities from the massive amount of various entities appearing in microblogs. To tackle the task, we introduce a new definition of emerging entities by focusing on how they appear in contexts that suggest their novelty. To collect such entities and contexts, we propose a novel method of distant supervision called time-sensitive distant supervision, which utilizes entities of knowledge bases and their timestamps of appearance in microblogs and develop an entity recognizer using those data.

Discovery of disappearing entities

Entities have the birth and the end, and recognizing the end is also helpful for various applications. For example, it is important to know about the death of a famous person, the bankruptcy of a company, or the discontinuation of a facility as soon as possible to help users make future decisions and maintenance of knowledge bases. However, even famous knowledge bases like Wikipedia are slow to update, and we cannot catch up with those disappearances by relying on those language resources. We thus aim to detect such disappearing entities from microblogs as soon as possible. Unlike emerging entities, it is challenging to apply time-sensitive distant supervision to disappearing entities because the timing of their occurrence is not clear. In addition, the number of disappearing entities is smaller than that of emerging entities, and the amount of training data does not scale, making it difficult to train the entity recognizer robustly. We thus improve the method of time-sensitive distant supervision for disappearing entities by explicitly considering the year of disappearance and collect high-quality training data. To train an entity recognizer robustly, we refine pretrained word embeddings using multiple microblog posts other than the input post and feed them as the additional input.

Typing of emerging entities

Even if we successfully detected emerging entities, it is difficult to extract associated knowledge and utilize it for subsequent applications due to the lack of occurrences in existing corpora or corresponding entries in knowledge bases. Therefore, we try to assign a type of entity such as baseball player, movie, and video game to emerging entities discovered in microblogs. The challenge is to perform typing of entities from short, noisy posts of microblogs without relying on linguistic resources such as knowledge bases, and how to deal with homographic entities, which share the same namings with existing entities (*e.g.,* 'Go' for a board game, a programming language, and a verb). To deal with noisy microblog posts, we develop a modular typing model that encodes not only contexts and entities but also microblog-specific meta-information from multiple posts. Furthermore, by introducing a context selector that selects only posts related to the target emerging entity from multiple noisy posts, we achieve robust prediction not only for non-homographic but also for homographic emerging entities.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Monitoring Microblogs through Entities

In recent years, with the development of smartphones and the internet, all kinds of people have come to communicate in cyberspace. In particular, a microblog, one of the social networking services such as Twitter, allows people to send out information in short posts whenever and wherever they are, and a vast amount of information from events in the real world to the private affairs of people is sent out in real-time [8]. Nowadays, users are not limited to individuals, but companies, local and national governments, are also actively utilizing the service for sending out official information [82, 37].

The information posted to the microblog has a variety of values and is useful to society in various ways, such as for people to track information on their interests, for companies to understand the trends of potential customers and their competitors, and for mass media and governments to understand local information in the event of incidents or disasters. This understanding of information through microblogs is generally referred to as social listening [17]. However, the amount and speed of information spread through microblogs are increasing [8], and it takes much effort to find and organize useful ones from a massive amount of posts.

Here, by focusing on entities such as works, locations, and events that appear in the text and recognizing them appropriately, we can organize their information in entity units even from a huge volume of posts and utilize them for social

listening applications such as social trend analysis, which requires candidates of what will capture the users' attention next. One of the most useful entities is the emerging entity, which is truly new and keeps appearing in the world, such as new products, works, companies, and events [30]. Monitoring these emerging entities and understanding their status in real-time can lead to understanding the trends in the real world. For example, people explore new works and events of interest, and social listening companies that analyze customer reputation, such as Oracle and Salesforce, monitor trends of new products, including those of their competitors.[1]

The counterpart to the emerging entities, disappearing entities, which are ceasing to exist from the world, such as people who have died, stores that are closing or going out of business, and products or events that are being discontinued, also play an important role in decision making about entities. For example, early recognition of the end of a work or event that people are interested in, or the discontinuation or end of support for a product, can help prevent loss of opportunity. Also, recognizing the bankruptcy of a company or the closing of a facility or store can help companies understand the situation in their industry.

In addition, to organize the information of discovered entities, it is necessary not only to find entities but also to estimate and give category/class information of the entity, such as baseball player, company, movie, and building at the same time [73]. This will allow us to group entities by types for focusing on entities with the specific type. This is especially important when targeting emerging entities that cannot refer to corresponding entries in the knowledge base for extracting knowledge.

1.2 Research Challenges

One way to monitor entities is to prepare an entity list and check the appearance of each entity on microblogs. Large-scale knowledge bases such as Freebase [13] and DBpedia [9], which have accumulated a large amount of information on entities, can serve as a reference for those entities. However, since these knowledge bases rely entirely on human labor to register entities and to describe specific

[1] https://www.salesforce.com/jp/products/marketing-cloud/social-media-marketing/

information about them, it is difficult to handle newly appeared emerging entities and also lack comprehensiveness of knowledge about the existence of disappearing entities. In addition, since the knowledge base is based on the standard that only notable entities with a certain degree of fame are registered[2], there are no entries for infrequent but valuable long-tail entities such as rising stars, local events, and stores. Therefore, it is necessary to recognize emerging and disappearing entities appearing in microblogs as soon as possible without relying on specific resources such as knowledge bases and regardless of the notability and frequency of the entities, and to utilize them for social listening applications.

To date, information extraction (IE), a technique for extracting structured data from unstructured or semi-structured documents, has been actively studied [18]. One of the major IE methods called named entity recognition (NER) [53] recognizes entities in text and attaches the type. However, NER models with usual training data are specialized in recognizing entity tokens in text [10] and cannot grasp the state of emerging or disappearing entities that have newly appeared or disappeared from the world. In addition, since those NER models tend to assign a label to each token based on the surface of the tokens in the input text, it overlearns token sequences that frequently appear in the training data [22]. As a result, it can recognize known token sequences with high accuracy while not generalizing to other infrequent or unknown token sequences. This also leads to false predictions regardless of the context when the target text has homogprahic entities (*e.g.,* 'Go' for a board game, a programming language, and a verb), which surfaces appeared in the training data. Therefore, it is difficult to apply NER as-is to the recognition and typing of emerging and disappearing entities, including many unknown or long-tail entities and homographic ones.

Although microblogs are ideal for IE because of the wide range of topics they cover and their real-time nature, they have the following two issues [21]. 1) The enormous volume of posts makes it difficult to collect useful posts for training and testing the model for any given task. 2) The character limit per post is short (*e.g.,* 280 characters in Twitter), and many of the posts are colloquial and noisy, containing emoji, hashtags, and URLs so that it is difficult to analyze with conventional IE techniques. To fully benefit from microblogs, it is necessary to

[2]For example, Wikipedia, one of the most famous knowledge bases, has registration criteria https://en.wikipedia.org/wiki/Wikipedia:Notability.

collect useful posts properly and to perform robust entity recognition and typing even under such circumstances.

1.3 Approaches and Contributions

Considering the above issues, we focus on a context in which the entity appears. When emerging and disappearing entities newly appear or disappear from the world, they are accompanied by the specific context. For example, contexts of emerging entities such as "The Matrix Resurrections" and "Torch Tower" contain expressions like 'ready for,' 'premiere,' 'trailer' and 'announce,' and disappearing entities such as "Daft Punk" and "Isamu Akasaki" contain expressions like 'death,' 'break up,' 'sad' and 'RIP.' These contexts often contain the necessary knowledge to convey information to potential readers, and we can utilize them for entity recognition and typing. Therefore, if we properly capture these contexts, *i.e.*, emerging and disappearing contexts, we can accurately and immediately recognize corresponding emerging and disappearing entities that appear at the same time. To collect these contexts efficiently on a large scale, we propose time-sensitive distant supervision, which utilizes Wikipedia entities and microblog timestamps.

In this case, if we simply train the NER model only with the emerging or disappearing contexts, the model will specialize on the entities that appear in the training data since it overfits to the sequence of the input sentence [22]. Therefore, we collect the same number of non-emerging and non-disappearing contexts using microblog timestamps for each of the collected emerging and disappearing contexts. By feeding these non-emerging and non-disappearing contexts separately from the corresponding emerging and disappearing contexts into the NER or typing model, they learn to recognize emerging and disappearing entities by distinguishing their contexts.

In the following, we summarize the tasks and contributions we have made based on the above approaches:

1. **Discovery of emerging entity**: We proposed a task for the early discovery of emerging entities from microblogs. To collect candidates of emerging entities and their emerging and non-emerging contexts efficiently on a large

scale, we targeted Twitter and proposed time-sensitive distant supervision that exploits Wikipedia entities and their microblog posts with timestamps. We developed a NER model with the collected data and applied it to Japanese and English Twitter archives. As a result, our method discovered emerging entities with higher precision than the baseline, which find unseen entities as emerging entities with burst detection. Furthermore, those discovered entities include not only notable emerging entities but also long-tail and homographic emerging entities. To evaluate relative recall, we tried to find emerging entities registered in Wikipedia from the Twitter archive. Our method successfully found more than half of the target emerging entities, and some of them were discovered on average one year earlier than the registration in Wikipedia.

2. **Discovery of disappearing entity**: We proposed a task for the early discovery of disappearing entities from microblogs. To collect disappearing contexts, whose timing of occurrence is not clear unlike emerging contexts, we provided time-sensitive distant supervision with temporal information of the disappearance from Wikipedia and constructed high-quality training data. In addition, to deal with the problem of unstable learning of the NER model due to the small number of disappearing entities in the training data, we refine pretrained word embeddings using the Twitter stream of the day of the input post and feed them to the model for robust entity recognition. Our method accurately recognized disappearing entities in Twitter posts than the baseline, which learn the data constructed with time-sensitive distant supervision without temporal knowledge. Same as the emerging entity, our method successfully found more than half of the target disappearing entities in Wikipedia, and some of them were discovered on average one month earlier than the update of the disappearance in Wikipedia.

3. **Typing of emerging entity**: We proposed a task to assign types such as baseball player, movie, or video game to emerging entities detected in microblogs. To perform entity typing from short and noisy posts in microblogs without relying on language resources such as knowledge bases, we focused on the phenomenon that emerging entities tend to appear in a burst of posts. We proposed a modular typing model that predicts the

type of entity from those multiple posts by encoding not only contexts and an entity surface but also meta-information specific to microblogs. In addition, to deal with homographic entities, which share the same namings with existing entities (*e.g.,* 'Go' for a board game, a programming language, and a verb) and thus are contaminating contexts, we introduced a context selector that selects the related emerging contexts of the target entity from the burst of posts. We input these posts into the typing model to achieve robust prediction for the type of entity containing homographic entities. We created a dataset for typing emerging entities by collecting emerging entities with the entity type and contexts from Twitter archives using time-sensitive distant supervision. Our proposed model achieved higher accuracy than the baseline model that randomly selects contexts during training and testing and does not use meta-information. For homographic emerging entities, our model outperformed the baseline by properly selecting contexts related to the target emerging entity.

1.4 Thesis Structure

This thesis is structured as follows. In Chapter 2, we first explain preliminary knowledge of core technologies that our approaches are based on. The following three chapters present the details of our approaches to the challenges mentioned above. In Chapter 3, we focus on emerging entities appearing in microblogs and propose time-sensitive distant supervision for collecting characteristic contexts of emerging entities. In Chapter 4, to discover disappearing entities, which are fewer in number than emerging entities, we improve the method of time-sensitive distant supervision and propose a method of refining pretrained word embeddings to incorporate features of multiple posts in an entity recognizer. In Chapter 5, to correctly predict the type of emerging entities, including homographic entities, we develop a modular entity typing model that encodes not only the entity surface and contexts but also meta-information of the posts, and a context selector that selects useful posts for typing of the target emerging entity. Finally, we present a conclusion of this thesis in Chapter 6.

Chapter 2

Preliminary Knowledge

The mainstream of natural language processing (NLP) tasks has been models based on highly representational neural networks. In this chapter, we describe the neural network-based machine learning methods in the field of NLP that we used in our study. First, we describe a basic network, the feed-forward neural network (FFNN), and then explain several methods for acquiring distributed representations of words *i.e.,* word embeddings [51]. We then describe the recurrent neural network (RNN) [24], and its variants for modeling text sequences and the conditional random field (CRF) [40] for sequence labeling of a sentence.

2.1 Feed-Forward Neural Network

We first introduce the most basic neural network, the feed-forward neural network (FFNN). FFNN basically consists of three layers: input layer, hidden layer, and output layer. In the input layer, the input sequence $\mathbf{x} = \{x_0, x_1, \cdots, x_L\}$ is given to the network, the hidden layer transforms $\mathbf{x}$ with the matrix W consisting of the weights of each element,[1] and the output layer transforms the last hidden layer with the weight matrix W to obtain the output value $\mathbf{y}$. L denotes the number of the input sequence.

[1]The model that iteratively performs this matrix transformation of the hidden layer is also called multi-layered perceptron (MLP).

This process can be formulated as follows:

$$\mathbf{h} = f_h(W_h\mathbf{x} + \mathbf{b}_h) \tag{2.1}$$

$$\mathbf{y} = f_y(W_y\mathbf{h} + \mathbf{b}_y) \tag{2.2}$$

Here, W_* and $\mathbf{b}_*$ are the weight matrix and bias vector, respectively, and are tuned by backpropagation. $f_*(\cdot)$ is an activation function for a vector $\mathbf{z}$ of dimensionality d that performs the nonlinear transformation of each element. The activation function often used in the hidden layer includes sigmoid function:

$$f(\mathbf{z})_i = \sigma(\mathbf{z})_i = \frac{1}{1 + \exp(-z_i)} \tag{2.3}$$

hyperbolic tangent:

$$f(\mathbf{z})_i = \tanh(\mathbf{z})_i = \frac{\exp(z_i) - \exp(-z_i)}{\exp(z_i) + \exp(-z_i)} \tag{2.4}$$

and rectified linear unit (ReLU):

$$f(\mathbf{z})_i = \mathrm{ReLU}(\mathbf{z})_i = \max(0, z_i) \tag{2.5}$$

For the activation function in the output layer, the following softmax function is often used:

$$f(\mathbf{z})_i = \mathrm{softmax}(\mathbf{z})_i = \frac{\exp(z_i)}{\sum_{j=1}^{d} \exp(z_j)} \; 1 \le j \le d \tag{2.6}$$

This transforms the output value into a probability distribution. $f(\mathbf{z})_i$ denotes the value of the i-th dimension of the vector $\mathbf{z}$ input to the activation function.

2.2 Word Embeddings

In the basic model of the NLP task, text data is input as a list of one-hot vectors, where each element corresponds to a word ID. However, one-hot vectors are costly to handle by computer because the number of dimensions becomes larger as the vocabulary size increases. In addition, it is difficult to calculate the inner product, which is necessary for calculating word similarities. Therefore, in many

cases, matrices that transform one-hot vectors into continuous vectors *i.e., word embeddings*, which are easy to handle on the computer, are acquired by learning them in advance or on the fly in the process of solving NLP tasks with neural network models. In this section, we describe methods for constructing word embeddings used in our study.

2.2.1 Skip-gram

Based on the idea of the distributional hypothesis [26], which states that the meaning of a word is based on the distribution of its surrounding words, Mikolov et al.[51] developed a neural network called Skip-gram for obtaining word embeddings from a large text corpus. Let $\{w_1, w_2, \cdots, w_C\}$ be the series of words of each sentence in the text corpus and for each sentence, the context $C_t = \{w_{t,1}, w_{t,2}, \cdots, w_{t,C}\}$ is the surrounding words of the target word w_t. Skip-gram maximizes the following probability by solving the task of estimating each word of the context from w_t:

$$P(w_{t,1}, w_{t,2}, \cdots, w_{t,C}|w_t) \tag{2.7}$$

The assumption is made that a word vector that maximizes this probability is a good representation of the word. Here, to calculate the degree of co-occurrence of w_t and $w_{t,i}$, they adopt inner product as score function. Then, the probability function is redefined as follows:

$$P(w_{t,1}, w_{t,2}, \cdots, w_{t,C_t}|w_t) = \prod_{c}^{C_t} \frac{\exp(\mathbf{v}_t^\top \mathbf{v}_{t,c})}{\sum_{c' \in W} \exp(\mathbf{v}_t^\top \mathbf{v}_{c'})} \tag{2.8}$$

W is the vocabulary. $\mathbf{v}_t$ and $\mathbf{v}_{t,c}$ are the word embeddings assigned to the words w_t and $w_{t,i}$, respectively. Note that this can be seen as a 3-layered FFNN that predicts the words of the context from the input word where W_h in (2.1) and W_y in (2.2) are embedding matrices.

However, since this function requires computing the softmax function for all words in W, the calculation becomes more difficult as the corpus size increases. Therefore, Mikolov et al. approximated the softmax function using a method called negative sampling. Specifically, they considered the task of estimating the context word as a task of identifying whether the context word was generated

from the training data or not, and treated words that actually appear around the target word in the training data as positive examples and words that do not appear around the target word as negative examples. Using sigmoid function to represent the probability distribution of generating positive and negative examples, respectively, they finally get the following objective function:

$$J = \sum_{c \in C_t} \log(1 + \exp(-\mathbf{v}_t^\top \mathbf{v}_{t,c})) + \sum_{n \in N_t} \log(1 + \exp(-\mathbf{v}_t^\top \mathbf{v}_{t,n})) \tag{2.9}$$

N_t is a set of words randomly sampled from the training data. Since a small number of samples is sufficient for obtaining better word embeddings, training is efficiently fast.

2.2.2 GloVe

Skip-gram learns word co-occurrences in each sentence and cannot capture global occurrence outside of the sentence. Considering the situation, Pennington et al.[61] created a word co-occurrence matrix for the given corpus and obtained word embeddings via matrix factorization. They changed the objective function to the following least-squares method for co-occurrences to reconstruct low-frequency and unknown word pairs:

$$J = \sum_{i,j}^{W} f(X_{i,j}) \left(\mathbf{u}_i^\top \mathbf{v}_j + b_i + b_j - \log X_{i,j} \right)^2 \tag{2.10}$$

Here, $f(\cdot)$ is the function indicating the weight of the degree of word co-occurrence, $X_{i,j}$ is the element of the co-occurrence matrix. $\mathbf{u}_i$ and $\mathbf{v}_j$ are the word embedding of the context and target word, respectively, and b_i and b_j are the bias terms.

2.2.3 fastText

Previous models have been trained on a token unit, ignoring the morphological information of the word. For example, 'go,' and 'goes' are treated as different words in Skip-gram and GloVe despite having the same meaning. In addition,

those models cannot obtain word embeddings for unknown words that do not appear in the trained corpus.

To deal with those problems, Bojanowski et al.[12] learn word embeddings in the framework of Skip-grams taking into account information of subwords. They consider each word as a set of character n-grams. For example, for the word 'going' when $n = 3$, the set of n-grams is <go, goi, oin, ing, ng>. They add the word itself <going> to this set and then assign an embedding to each element. Finally, the embedding of each word is expressed as the sum of the embedding of each element. For this reason, they modify the function of Skip-gram, which measures the degree of co-occurrence between the target word and the context word as follows:

$$\sum_{g \in G_{w_t}} \mathbf{v}_g^\top \mathbf{v}_{t,c} \tag{2.11}$$

In this way, words can be represented compositionally by subwords, and thus we can consider morphemes for those embeddings. We can obtain the word embedding for unknown words not by the word itself but by the sum of its subwords.

2.3 Recurrent Neural Network

Since FFNN described in § 2.1 has a fixed input and output length, it is difficult to handle language data whose sequence length is variable, unlike image data. In addition, FFNN cannot maintain the order information of the input, which is important for modeling sequential data. In the field of NLP, the recurrent neural network (RNN), which is the neural network with internal states, is often used to handle language data. In recent years, modern architectures of RNNs, such as Long-short Term Memory (LSTM) [33] and Gated Recurrent Unit (GRU) [16], are especially used for better modeling of longer sequences. In this section, we first explain the most basic RNN called Elman Network [24], followed by LSTM, GRU, and bidirectional RNN [68], all of which are used in our study.

2.3.1 Elman Network

Elman Network [24] is the simplest form of RNN, which has a directed acyclic graph inside and can handle variable-length input sequences. Given a sequence $\mathbf{x} = (x_0, x_1, \cdots, x_L)$, it takes as input x_t at every time step t. L denotes the number of the input sequence. Internally, it keeps the hidden layer of the previous time $\mathbf{h}_{t-1}$ and uses it with the input to calculate the next hidden layer $\mathbf{h}_t$. The output layer takes the hidden layer to obtain the output value $\mathbf{y}$. The formula of RNN is as follows:

$$\mathbf{h}_t = f_h(W_h \mathbf{x}_t + U_h \mathbf{h}_{t-1} + \mathbf{b}_h) \tag{2.12}$$

$$\mathbf{y}_t = f_y(W_y \mathbf{h}_t + b_y) \tag{2.13}$$

W_*, U_*, and b_* are trainable weight matrices and bias vectors of the model, respectively. $f_*(\cdot)$ are nonlinear functions.

RNN performs backpropagation during training, and it propagates gradient back to the first input x_0. As a result, the gradient at time t is continuously multiplied during the calculation and eventually disappears. This gradient vanishing problem [32] becomes especially serious when the input sequence gets longer.

2.3.2 Long Short-term Memory

Long Short-Term Memory (LSTM) [33] is the network of RNN equipped with a memory unit and three gates, which stores and releases a certain amount of gradient in the memory unit through the gating mechanisms. This mechanism allows LSTM to handle longer input sequences than regular RNNs. The input gate $\mathbf{i}_t$ controls the flow of information from the input $\mathbf{x}_t$ and the hidden layer one step before $\mathbf{h}_{t-1}$ to be kept in the memory unit $\mathbf{c}_t$. Conversely, the forget gate $\mathbf{f}_t$ releases the information held in the memory unit $\mathbf{c}_t$. Finally, from the memory unit $\mathbf{c}_t$ and the output gate $\mathbf{o}_t$, the current hidden layer $\mathbf{h}_t$ is calculated as follows:

$$\mathbf{i}_t = \sigma\left(W_i\mathbf{x}_t + \mathbf{b}_{xi} + U_i\mathbf{h}_{(t-1)} + \mathbf{b}_{hi}\right) \tag{2.14}$$

$$\mathbf{f}_t = \sigma\left(W_f\mathbf{x}_t + \mathbf{b}_{xf} + U_f\mathbf{h}_{(t-1)} + \mathbf{b}_{hf}\right) \tag{2.15}$$

$$\mathbf{g}_t = \tanh\left(W_g\mathbf{x}_t + \mathbf{b}_{xg} + U_g\mathbf{h}_{(t-1)} + \mathbf{b}_{hg}\right) \tag{2.16}$$

$$\mathbf{o}_t = \sigma\left(W_o\mathbf{x}_t + \mathbf{b}_{xo} + U_o\mathbf{h}_{(t-1)} + \mathbf{b}_{ho}\right) \tag{2.17}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{(t-1)} + \mathbf{i}_t \odot \mathbf{g}_t \tag{2.18}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh\left(\mathbf{c}_t\right) \tag{2.19}$$

W_*, U_*, and b_* are trainable weight matrices and bias vectors for each layer of the model, respectively.

2.3.3 Gated Recurrent Unit

Gated Recurrent Unit (GRU) [16] is a network of RNN that is faster than LSTM and has similar performance while removing some of the gates in the LSTM to reduce the parameters and simplify the structure. GRU keeps the gradient in the hidden layer $\mathbf{h}_t$ instead of the memory unit and integrates the input gate and output gate in LSTM into the update gate $\mathbf{z}_t$. The reset gate $\mathbf{r}_t$ is used as the auxiliary to the update gate and it behaves like the forget gate in LSTM. The current hidden layer $\mathbf{h}_t$ is calculated using those two gates as follows:

$$\mathbf{z}_t = \sigma\left(W_z\mathbf{x}_t + \mathbf{b}_{xz} + U_z\mathbf{h}_{(t-1)} + \mathbf{b}_{hz}\right) \tag{2.20}$$

$$\mathbf{r}_t = \sigma\left(W_r\mathbf{x}_t + \mathbf{b}_{xr} + U_r\mathbf{h}_{(t-1)} + \mathbf{b}_{hr}\right) \tag{2.21}$$

$$\hat{\mathbf{h}}_t = \tanh\left(W_h\mathbf{x}_t + \mathbf{b}_{xh} + r \odot U_h\mathbf{h}_{(t-1)} + \mathbf{b}_{hh}\right) \tag{2.22}$$

$$\mathbf{h}_t = \mathbf{z}_t \odot \mathbf{h}_{t-1} + (\mathbf{1} - \mathbf{z}_t) \odot \hat{\mathbf{h}}_t \tag{2.23}$$

$$\tag{2.24}$$

W_*, U_*, and b_* are trainable weight matrices and bias vectors for each layer of the model, respectively. It is reported that GRU has worse memory performance for longer sequences than LSTM due to the smaller number of parameters.

2.3.4 Bidirectional RNN

Although the normal RNN inputs the series $\mathbf{x}$ in order from the first element x_0, RNN that also considers inputs in the reverse direction from the last element x_L is called bidirectional RNN [68]. This model can compensate for the information at the beginning of the series that the forward model tends to lose due to gradient calculation, and as a result, it improves the performance of the model than the normal RNN. Bidirectional RNN combines both the hidden layer of the forward RNN $\overrightarrow{\mathbf{h}}_t$ and the hidden layer of the backward RNN $\overleftarrow{\mathbf{h}}_t$ as follows.

$$\mathbf{h}_t = [\overrightarrow{\mathbf{h}}_t, \overleftarrow{\mathbf{h}}_t] \tag{2.25}$$

Of course, this technique can be applied to RNN variants such as LSTM and GRU.

2.4 Conditional Random Field

Consider a situation where we assign a label, such as a part-of-speech (POS) tag, to each element of an input sequence. Although the labels can be determined by fitting a softmax function to the output layer, if there are dependencies between the labels, it is problematic to perform estimation for each element individually (*e.g.,* in POS-tagging in English, the subject is almost always followed by a verb). Therefore, there is a method called conditional random field (CRF) [40] that takes into account the dependency between labels and determines the final output. Given an input series $\mathbf{x} = (x_0, x_1, \cdots, x_L)$ and a predictive label $\mathbf{y} = (y_0, y_1, \cdots, y_L)$, CRF defines a score $S(\cdot)$ as follows:

$$S(\mathbf{x}, \mathbf{y}) = \sum_{i=0}^{L} A_{y_i, y_{i+1}} + \sum_{i=1}^{L} P_{i, y_i} \tag{2.26}$$

A_* is the score for transitions between labels, and P_* is the prediction score for each label output by the model. For unlikely transitions, the score of A_* will be

lower. Using this score and the softmax function, the label series $\mathbf{y}$ given the input series $\mathbf{x}$, can be expressed as follows:

$$p(\mathbf{y}|\mathbf{x}) = \text{softmax}(S(\mathbf{x}, \mathbf{y})) \tag{2.27}$$

Finally, the objective function is expressed as follows:

$$J = \log p(\mathbf{y} \mid \mathbf{x}) = S(\mathbf{x}, \mathbf{y}) - \log \sum_{y' \in \mathbf{y}} \exp(S(\mathbf{x}, \mathbf{y}')) \tag{2.28}$$

This is optimized by Viterbi algorithm [28], a kind of dynamic programming. CRF can be applied to RNN and its variants as a CRF layer.

Chapter 3

Discovery of Emerging Entity

3.1 Introduction

Understanding the latest world events is an important objective for many applications such as social-trend analysis, marketing research, and reputation management. Such applications often require knowledge of *emerging entities*, such as new products, works, devices, and individuals, which ceaselessly emerge one after another for real-time monitoring of their activities. For example, social listening companies such as Salesforce and Oracle that monitor customer reputations need to track new product trends, including customers' competitors. People are also constantly trying to keep up with emerging entities in their genre of interest, such as new songs, devices, and events. Although knowledge bases (KBs) such as Wikipedia could be used as a reference list of entities, there is an inevitable delay until the emerging entities are registered in those KBs, and only notable entities are selected for registration. Therefore, instead of relying on KBs, we need to discover as many emerging entities as possible, including *long-tail (less frequent but wide variety of) emerging entities* that are mostly overlooked in KBs before they become prevalent or their information appears frequently.

One possible solution for handling emerging entities is to monitor news or microblogs and collect all named entities (NEs) detected through named entity recognition (NER). However, NER just recognizes mentions of NEs in a text [10, 22], it cannot notice *homographic emerging entities* that share the naming with existing entities (*e.g.*, "Go" refers to an emerging programming language

and a board game). Although entity linking [15] disambiguates the entities for their mentions in the text, it can handle only entities registered in KBs.

Some of the previous studies [54, 34, 75, 25] focus on detecting out-of-KB entities which are not registered in a particular KB and consequently find massive non-emerging entities since the absence of the entities in KBs does not guarantee their emergence. Extracting emerging entities from the obtained out-of-KB entities is difficult since out-of-KB entities are mostly mere long-tail entities, and we cannot expect many contexts to judge their emergence. The contexts can even be noisy when they are homographic emerging entities. Even worse, it is problematic to prepare training (and evaluation) datasets for out-of-KB entities since we need to manually annotate them that are not registered (but notable[1] enough to register) based on the specific state of the given KB.

Considering these difficulties, we introduce a novel task of discovering emerging entities in a microblog when they have just been introduced to the public through the microblog. This task is more solid than the existing out-of-KB entity classification task since the task definition is independent of a particular KB. In our task, we use the fact that people write about emerging entities with *expressions suggesting their emergence and types* when those entities are not well known to the public (Table 3.1), considering that potential readers would be unfamiliar with them. By taking advantage of these contexts, we can effectively discriminate emerging entities from others, even if they are long-tail or homographic emerging entities, and can find them in the early stage of their appearance.

To obtain contexts of emerging entities, we propose a time-sensitive distant supervision method based on distant supervision [52]. Our method collects early-stage posts in a massive amount of time-series text where non-homographic entities registered in a KB first emerge. At this time, to discover emerging entities, we collect adequately-later posts after the first appearance as negative examples to robustly discriminate them from emerging contexts. We then train sequence-labeling models from those contexts to discover emerging entities.

We applied our method to our large-scale Twitter archive to discover English and Japanese emerging entities and compared the discovered entities with those obtained with Baselines, which regard entities that are unseen in a KB [54] or

[1]https://en.wikipedia.org/wiki/Wikipedia:Notability

Nintendo *announces* **Super Smash Bros. Ultimate** - the game features every single character from past games. Oh, and it has GameCube controller support. *Release data: December 7, 2018* URL
JohnnyDepp is set to play celebrated war photographer W. Eugene Smith in *upcoming drama* "**Minamata**" which HanWay Films *will launch at the upcoming* AFM. Film director is Andrew Levitas, Based on the book by the same name, *Filming starts in Japan then Serbia in January 2019* URL
UP-FRONT Group have *announced* the *formation of* **College Cosmos**, a 25 person idol group of university students, formed with Space Craft Group who run college beauty pageants, "combining beauty and intelligence". Country Girls *member* **Risa Yamaki** also *joins the Group*.
A behind the scenes look at **EvermorePark** which looks like it's *going to be a truly magical place!* Check it out! URL
New '**Keytar Bear**' beer from Trillium will raise money for injured performer URL HASH HASH
Can't wait for this one. **Big Dom's Bagel Shop** *will open* Aug. 25 in Cary. Here are all the details on Pizzeria Faulisi getting into the bagel business URL
Talented music group La Meme Gang *is set to* treat students of the University of Professional Studies, Accra to an exciting Performances *come 3rd November 2018 from 1pm to 8pm.* The group *will perform* **Club Shandy Block Out Party**.

Table 3.1: Example tweets on emerging entities (bold) with expressions suggesting their emergence (italic).

our Twitter archive as emerging. Experimental results showed that the proposed method effectively detected emerging entities in terms of precision of the acquired entities, including homographic and long-tail emerging entities. As the evaluation of relative recall and detection immediacy, using the entities newly registered in Wikipedia as a reference, our method detected most entities in the reference, and in most cases, these entities were discovered earlier than their registration in Wikipedia.

Our contributions are as follows:

- We introduce a novel task of discovering emerging entities in microblogs as early as possible.

- We propose a time-sensitive distant supervision method for efficiently and automatically constructing a large-scale training dataset from microblogs.
- Our method found emerging entities accurately (high precision), abundantly (high recall), and quickly (substantially earlier than their registration in Wikipedia).
- We will release all the datasets (tweet IDs)[2] used in experiments to promote the reproducibility.

3.2 Definition of Emerging Entity

In this section, we define what is meant by the term *emerging entity* in this study. Our definition of an emerging entity is motivated by the report of Graus et al. [30] and meets requirements for social listening applications.

Graus et al. analyzed how newly registered entities in Wikipedia have appeared in the news and social media before they are registered as individual articles. They found that most of those entities shift from the state of "sporadically mentioned in the news and social media" to that of "established as one article due to enhancement of references."

Fortunately, when users submit posts about entities that appeared newly but are not famous yet to social media, they usually indicate the emergence and the characteristic of the entities, as in Table 3.1, despite their popularity. We thereby define emerging entities in terms of how they are described in contexts, in other words, how their state is perceived by people as follows:

Emerging contexts. *Contexts in which the writers assumed the readers do not know the existence of the entities.*

Emerging entities. *Entities in the state of being still observed in emerging contexts.*

We also define other terms on entities as follows:

Prevalent contexts. *Contexts in which the writers assumed the readers know the existence of the entities.*

[2]http://www.tkl.iis.u-tokyo.ac.jp/~akasaki/ijcai-19/

Prevalent entities. *Entities in the state of being mainly observed in prevalent contexts.*

Long-tail entities. *Entities that are less frequent individually but have wide varieties.*

Homographic entities. *Entities that share the namings with other entities.*

Here, long-tail entities are usually difficult to detect because they are low-frequency, and thus clues cannot be obtained on a large scale. Also, homographic entities are required to be handled separately by a method like word sense disambiguation [55] because their contexts are mixed with that of other entities with the same surface. Note that even for these entities, we still observe specific state transitions described above when they first emerge. We can, therefore, immediately recognize them by capturing the emerging context of the entity, regardless of its frequency or polysemy, and without any special treatment. To validate the solidness of these definitions, we evaluate the inter-rater agreement of emerging entities acquired from the text and also confirm that we can find many long-tail and homographic entities by exploiting the emerging contexts (§ 3.5.4).

3.3 Related Work

To the best of our knowledge, there has been no study attempting to find emerging entities in microblogs. We review the current tasks related to our task and clarify the term "emerging entities," which has various meanings.

3.3.1 Emerging and Rare Entity Recognition

This is a task organized at the 2017 Workshop of Noisy User-generated Text (WNUT 2017) [22] and focused on recognizing both "emerging and rare" entities from text. With this task, named entities (NEs) that appeared zero times in specific (past) portions of datasets are regarded as emerging entities, and manually annotated NE tags to these entities as the target of detection regardless of the contexts in which they have appeared. The dataset used in this task includes the following examples (the target entities are in bold):

> ... found photo storage tank that is 5x size of my **iPhone** with less capacity than **iPhone 4** ...
>
> ... Woke up in **Sacramento**. The **CA** weather feels great. Good workout. Good **Subway** sandwich ...

Consequently, this task is designed to detect (past) data-dependent emerging entities even after they become known to the public (*e.g.,* iPhone and Subway). The definition of emerging entities based on specific data makes it difficult to distinguish emerging entities from prevalent entities. In fact, the state-of-the-art model achieved an F_1 of 49.59% [72], which is much lower than usual named-entity recognition (NER) on a dataset such as CoNLL-2003 [67] (F_1 of 94.6%) [71]. Our task discovers emerging entities when they are introduced in microblogs. This enables us to take advantage of the fact that emerging entities tend to show their emergence at the early stage of their appearance.

3.3.2 Out-of-KB Entity Identification on News Articles

This task has been studied to identify NEs that are not registered in a KB (referred to as "emerging" entities in the following studies but as out-of-KB entities here for clarity). Since this task is intended to detect entities absent in the KB, it does not distinguish emerging entities from mere long-tail entities. Nakashole et al. [54] proposed a method for extracting NEs using NER and regards all extracted NEs as out-of-KB if they are not registered in a KB. Since this method ignores contexts in which NEs appear, if the target NE has homographic entities in the KB, it is wrongly classified as an in-KB entity regardless of its emergence (false negatives). Similarly, if the target NE appears with the unseen surface (mention), it is wrongly classified as an out-of-KB entity (false positives).

Hoffart et al. [34], Wu et al. [75], and Färber et al. [25] proposed methods of classifying whether a given NE in a news article is out-of-KB. Their task is part of the task solved by Nakashole et al. [54] since the target NEs are given (assumed to be recognized). Note that NEs are, however, not easily recognizable for languages in which NEs are not capitalized (*e.g.,* German, Chinese, and Japanese). In addition, their methods do not scale to ever-increasing emerging entities because the manual annotations of out-of-KB entities depend on the

specific state of the KB, and the approaches (and features of the classifier) are tailored for news text. In contrast to these studies, we focus on "truly" emerging entities defined independently of KBs and develop an early-detection method using a dataset constructed by time-sensitive distant supervision. We targeted microblogs, *i.e.,* timely social media, as sources for emerging entities since Graus et al. [30] reported that emerging entities appear on social media more and earlier than in news articles.

3.3.3 Notable Account Prediction on Twitter

This task is to discover long-tail "rising" entities (*e.g.,* rising brands) that are expected to be notable in the future within Twitter [14]. Although this task uses Twitter as the source of entities, the same as ours, it requires experts to provide examples of notable entities. Also, since the target entities are limited to only those with Twitter accounts, it cannot acquire various types of entities that are not linked to Twitter accounts. We also focus on Twitter but discover emerging entities (§ 3.2) without relying on domain experts and without restricting the types of entities to be discovered.

Overall, these related studies defined labels (emergence or rare, out-of-KB, or notability) based on specific past data, KBs, or domain experts and annotated them manually. We compare our method with two Baselines that detect unseen NEs in a KB [54] or in the past Twitter (the same setting as WNUT17) as emerging. We chose these methods because they are the only methods applicable to our task, which do not rely on manually annotated data.

3.4 Proposed method

The proposed method discovers emerging entities in microblogs. We target a microblog (Twitter) since Graus et al. [30] reported that compared to news articles, a more diverse range of emerging entities appear earlier on social media, and generally speaking, microblogs include the most timely posts among various types of social media. Note that we do not exploit Twitter-specific functions with our method; thus, it is also applicable to other microblogs such as Weibo.

To build a supervised model for discovering emerging entities, we exploit the fact that emerging entities are likely to appear in specific contexts (§ 3.2). By properly identifying such contexts, we can discover and type corresponding emerging entities effectively and instantaneously, even if they are long-tail or homographic ones. The major challenge lies in how to collect such emerging contexts as the training data. To cover various emerging contexts for a diverse range of entity types, we develop a method that automatically collects such contexts and corresponding emerging entities.

3.4.1 Time-sensitive Distant-supervision

To meet the expected requirements on the training data for this task, we developed the proposed method (Figure 3.1) based on time-series text and the distant supervision [52], which automatically collects training data using an existing KB for a specific knowledge-acquisition task. Since our method does not incur any annotation cost, it is easy to prepare and construct the training data. The major difference from the original distant supervision is that labels are not defined only with the KB. We utilize the nature of time-series text to obtain labels for training an emerging entity recognizer.

The idea is to first extract non-homographic entities with unique namings from a KB that emerge when microblog posts are available and to collect their emerging contexts from the time-series microblog posts. The procedure is as follows:

Step 1 (Collecting candidates of emerging entities)

We start by collecting titles of articles in Wikipedia as existing entities and then associate them with the time-stamps of registration to collect emerging entities that newly appeared within the available period of the microblog (Twitter). We exclude entities that appeared on Twitter more than i times in the first one-year period where microblog posts are available. This is to exclude homographic entities that share the naming with prevalent entities since it is difficult to collect their emerging contexts only by searching the entities.

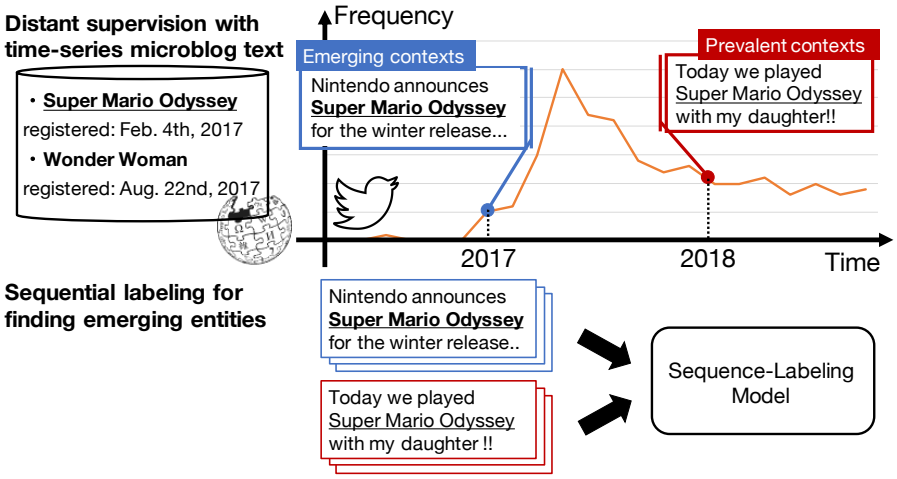

Figure 3.1: Time-sensitive distant supervision: for the entities retrieved from a KB, emerging and prevalent contexts are collected from microblogs, and sequence labeling models are trained from the obtained emerging and prevalent contexts.

Step 2 (Collecting contexts of emerging entities)

For each entity obtained in Step 1, we then retrieve the first k early-stage microblog posts posted before the time-stamps of registration as emerging contexts. Although contexts of long-tail emerging entities are not covered in the obtained training data, similar emerging contexts can be shared by other entities in the KB. This is because if the coarse type of entities is the same, their emerging contexts tend to be common regardless of their popularity (*e.g.*, product types tend to be introduced with the term *released*).

There are two issues to be addressed: 1) how to filter noisy examples of emerging contexts and 2) how to prevent overfitting that detects only the entities used for collecting training data. We explain how we address these issues.

Filtering Noisy Emerging Contexts

Although distant supervision can generate abundant training data, incorrectly labeled data can also be included. We, therefore, collect only reposts (retweets) from the day when the included entities first appeared more than i' times in retweets. This is inspired from the report of [30] that emerging contexts are likely to be shared by many users since they include information novel to the public.

Collecting Prevalent Contexts as Negative Examples

When a model is trained only with the collected emerging contexts, it will be overfitted to detect only mentions of the emerging entities used to collect the training data. To avoid this, in Step 2, we collect prevalent contexts for the same entities collected in Step 1 as negative examples (Figure 3.1). Specifically, as the prevalent contexts for each entity, we collect the same number of microblog posts one year after the time of collecting emerging contexts. This enables the model to discriminate between emerging and prevalent contexts and reduces the effect of noisy (prevalent) contexts incorrectly included in the positive examples.

We finally label only the acquired entities in the emerging contexts as emerging entities and combine them with their prevalent contexts to form the training data for sequence labeling described below. We tried several values for the three hyperparameters of our method, i, k, and i', and confirmed that the accuracy of the models trained from the resulting training data did not markedly change. We therefore empirically set the parameters to $i = 5$, $k = 100$ and $i' = 10$.

3.4.2 Sequence Labeling for Finding Emerging Entities

We next train a sequence-labeling model for finding emerging entities from the collected training data. We adopted modern long short-term memory (LSTM) with CRF output layer (LSTM-CRF) [41] as the sequence-labeling models. LSTM-CRF inputs a word embedding and character embeddings encoded by character LSTM of each token into bi-directional LSTM, which are followed by the CRF layer.

We adopted BIOES as the tagging scheme, which was reported to be better than other schemes [63]. We tagged emerging entities in positive examples with BIES and the others with O.

3.5 Experiments

We applied the proposed method to the actual Twitter archive and performed our task of discovering emerging entities.

TYPE DBpedia types	# entities	# posts	examples of emerging context (truncated)
PERSON	**9348**	**69427**	
Person	2352	14415	[KR NEWS] 141129 **Ko Sung-hee**, confirm to appear in 'SPY'.. acting with Kim Jaejoong URL
SoccerPlayer	1194	7346	NEW VIDEO! Legia's 16 year old **Krystian Bielik**, MEGA talent! URL Enjoy! Share!
AmericanFootballPlayer	948	8747	UGA junior DL **Kwame Geathers** to declare for NFL Draft URL via URL
Politician	941	7316	... confirms earlier scoop by _USER_: **Abdi Farah Shirdon Saaid** is the new Somali Prime Minister
BaseballPlayer	535	4948	Congrats to **Brett Marshall**, who seems almost certain to make his big-league debut tonight.
BasketballPlayer	345	3062	The fourth pick of the 2015 PBA Rookie Draft Meralco Bolts select **Chris Newsome**! HASH
IceHockeyPlayer	323	3313	Carolina cuts deficit to 4-1 in second period on **Jaccob Slavin** goal. It is his first NHL goal.
Cricketer	273	1991	**Jake Libby** is the first Nottinghamshire player to score a century on his Championship debut for ...
RugbyPlayer	253	1761	Sydney's newest Rooster **Kane Evans** caught up with HASH after a stellar debut. Check it out ...
Others (63 types)	2184	16528	Massive congratulations to **Marjorie Celona** for being picked for the HASH for her debut 'Y' ...
CREATIVE WORK	**6599**	**47930**	
Film	1622	9078	The official title for Episode VII is '**Star Wars: The Force Awakens**.'
Album	1308	9086	_USER_ return with new album **The Scene Between**. Hear the title track URL
TelevisionShow	979	5879	See your favorite stars of HASH and HASH in '**Bachelor in Paradise**', coming this summer ...
VideoGame	898	10371	**The Last of Us Remastered** PS4 pretty much confirmed URL
Single	700	6544	Gwen Stefani's comeback single (yes, gawd!) '**Baby Don't Lie**' is out on 6 October URL
Book	339	1586	The cover of my book is out! **Crazy Town: The Rob Ford Story**. It's being released a month ...
Song	139	1735	Watch Beyoncé share her NEW song "**Die with You**" for her & JAY Z's 7th anniversary:
Manga	135	652	HASH Kodansha Comics Adds Hiro Mashima's **Fairy Tail Zero** Prequel HASH
Software	134	1067	We've got big announcements: _USER_ support. Introducing **Kubernetes**.
Others (16 types)	345	1932	...write notes? Capture ideas? Work with other people? Then sign up for the **Dropbox Paper**.
LOCATION	**1326**	**5729**	
Bank	515	2616	Tencent ready to launch China's 1st private internet bank, **WeBank**
City	311	953	Some fantastic photos from the opening of the new **Kerry Town** HASH treatment centre ...
Hotel	68	280	Privileged to attend **Dr Chau Chak Wing Building** Launch Dinner tonight at UTS- a ...
Building	55	186	89th and Western, **Oak Crest** Church of Christ is opening its doors for folks whose homes were ...
Stadium	52	404	... The Hartford Yard Goats just named their new ballpark in HASH "**Dunkin' Donuts Park**."
School	44	214	Exciting day with HRH Duchess of Cambridge opening HASH **Aldridge Academy** this ...
Others (30 types)	320	1304	South Africa will have two new universities, they will be the **Sol Plaatje University** in the ...
GROUP	**644**	**4027**	
Organisation	193	1315	Very well done to _USER_, who is appointed first director of **Creative Industries Federation**. ...
Band	109	441	180514 Jackson weibo update **TFBoys** HASH HASH URL
PoliticalParty	61	463	Ozawa's new party's English name is **People's Life First** (LF), party sources told The Japan Times.
SoccerClub	54	429	... haven't realized is that the Sounders are moving to Oklahoma and plan to rebrand as **Rayo OKC**.
Company	42	243	**HitBliss** Launches Hulu Competitor That Turns Ads Into Currency URL
Others (18 types)	185	1136	Sudan's Janjaweed militia operating under a new name, **Rapid Support Forces** HASH URL
EVENT	**393**	**2508**	
Award	104	746	New literary award **The Folio Prize** launches as 'Booker without the bow ties' URL
MixedMartialArtsEvent	45	384	Dana White announces **UFC 171** in Dallas at the AAC. _USER_ needs to be on that card!
SpaceMission	42	238	[Video] _USER_ John Grunsfeld announcing our new **Mars 2020** mission early today at HASH
Convention	39	228	In other news, SXSW has spawned "**SXSW V2V**," a four-day event in Vegas in August (ugh). ...
MilitaryConflict	22	92	Syria's Assad plans **Aleppo offensive** as U.S. plans to arm rebellion seen as too little, too late URL
Others (16 types)	141	820	Every chance of another flying dismount after next race - **Dubai Gold Cup**. Dettori on board well ...
DEVICE	**312**	**3767**	
Device	145	1952	**iPhone 5c** will cost $99 for 16GB and $199 for 32GB. IPhone 5C is the newest addition to apple ...
Automobile	65	592	Is **Mazda CX-3** easily the best-looking entry in the new batch of compact SUVs? URL URL
InformationAppliance	62	997	YipRT _USER_: **Oculus Rift**: Step Into the Game by Oculus - Kickstarter URL via _USER_
Ship	26	132	Australia's new (& only) marine science ship **RV Investigator** arrives in Hobart home port. URL
Others (4 types)	14	94	IDF plans laser interceptor **Iron Beam** for short-range rockets - URL
OTHER	**498**	**3857**	
Owl:Thing	276	2177	**Mars One** will start recruiting volunteers in July for one-way trip to red planet URL
Horse	77	1030	Everyone aware one Kentucky Derby horse this year is called **Palace Malice**? URL
Holiday	34	229	**Transgender Awareness Week** starts today. Here's a poster to help you explain the event: URL
GivenName	26	59	STORM OF VOID is my new band. Me on 8 string guitar, **Dairoku** from "envy" on drums, ...
Others (18 types)	85	362	A compact version of our inner solar system has been found orbiting **Kepler-444**... but ...
UNMAPPED	11825	62108	PAGASA again explaining forecast for **Typhoon Ruby** (int'l name Hagupit) as NDRRMC ...
			As of 20 April Guinea has reported a cumulative total of 208 clinical cases of **Ebola virus disease** ...
Total	30945	199353	

Table 3.2: Statistics of the English emerging entities and their contexts obtained from our Twitter archive by our time-sensitive distant supervision.

TYPE DBpedia types	# entities	# posts	examples of emerging context (truncated)
PERSON	**4932**	**23939**	
Actor	885	2863	なしみかんの相方、ミカちゃんが『田所ミカ』から『**野々宮ミカ**』に改名いたしました。 ミカちゃん自身の...
MusicalArtist	731	4616	歌って!踊って!釣れるアイドル「**つりビット**」5・22デビュー(デイリースポーツ) - Y!ニュース URL
SoccerPlayer	531	2327	この度、**三吉聖王**選手が、清水エスパルスへ加入することが決定しましたのでお知らせいたします。URL
VoiceActor	484	1596	期待の新人声優「**雨宮天**」生放送で初の顔出し出演!超可愛いと話題に・・・ URL
BaseballPlayer	419	4390	【新外国人】DeNA、カブスの**スペンサー・パットン**を獲得へ URL URL
AdultActor	299	1731	今月SODからデビューした**本田莉子**ちゃんのデビュー作を観ているけど、メチャクチャかわいい...
Model	281	1343	初めまして!ジュノン・スーパーボーイ・コンテスト出身の**勧修寺保都**です。所属事務所が決まり...
Politician	260	1164	【川崎市長選 **福田紀彦**氏が当確】 任期満了に伴う川崎市の市長選挙は27日に投票が行われ、無所属の新人で...
Person	177	729	米国家安全保障局(NSA)の監視を暴露した内部告発者、**エドワード・スノーデン**氏のインタビュー動画。URL
Writer	152	542	【本日発売】新潮11月号。新潮新人賞発表されました。**高橋弘希**、指の骨。選評、川上未映子...
Wrestler	136	576	うおっ楽しみ!新人のデビュー戦の相手がWNC女子王者!**小林香萌**vsリン・バイロン!...
ComicsCreator	93	361	以前イブニングで新人賞の審査をさせていただいたときに受賞した**松浦だるま**さん...
Others (18 types)	577	2062	先に言っちゃいます。 **厚切りジェイソン**さんというお笑い芸人さんが登場するんですが...
CREATIVE WORK	**6460**	**47267**	
MusicSingle	1321	11685	12月23日発売 「**右足エビデンス**」 ソロシングルの曲名が発表されました!予約受付中です!
TelevisionShow	1153	8478	【大久保佳代子】本日からTBS新番組「**おーくぼんぼん**」スタート!24:50～25:20 MCに大久保佳代子と
MusicAlbum	970	6092	キスマイ アルバムタイトル 「**Kis-My-Journey**(キスマイジャーニー)(仮)」! まいどジャーニー> かよwwwwww...
Film	917	6307	...が登壇。また、ソニーの新スパイダーマン映画が「**スパイダーマン:ホームカミング**」に正式決定...
VideoGame	652	6355	【速報】『**DARK SOULS III**』発表! 2016年初頭に発売予定【E3 2015】 URL
Manga	623	2561	...されてましたが、来週の少年ジャンプ32号より堀越先生の新連載「**僕のヒーローアカデミア**」が始まります!
Anime	323	2983	京アニ制作 TVアニメ「**たまこまーけっと**」2013年1月放送開始! 公式オープン、スタッフ・キャスト...
RadioProgram	266	1010	【新番組!】STVラジオで10/12から毎週日曜24時『**藤岡みなみのおささらナイト**』がスタート...
Book	146	983	『**Re:ゼロから始める異世界生活**』書籍化おめでとうございます!! URL HASH
Song	32	174	「みんなのうた」10月～11月の新曲に畑亜貴さんの「**図書館ロケット**」がオンエアされます。URL
Others (4 types)	57	639	KADOKAWA×はてな 新小説投稿サイト名は「**カクヨム**」に決定!2月末にいよいよ本格始動!
LOCATION	**371**	**1554**	
Building	121	756	上野に新グルメビル「**上野の森さくらテラス**」誕生–ランチもディナーもおまかせ! URL
Museum	42	184	**熊谷守一つけち記念館**開館記念式典が始まりました。小南館長の挨拶 URL
Station	34	115	JR南武支線に設置される新駅の名称は、投票の結果、「**小田栄駅**」に決定...
Settlement	28	47	この人たち、ほんと自分たちのことしか考えてないでしょ。→夢の街**スラブチッチ**を福島に HASH URL
School	24	34	**開志国際高等学校**開校式に出席しました。ソチオリンピック銀メダリストの平野歩夢選手も新入生です。
City	17	46	承前)**ラハダトゥ**の街、戦闘の影響なのかわかりませんが、携帯電波が悪すぎ。ネットは問題ないです...
University	14	47	たいへんなことになりました。明後日11/4、**秋田公立美術大学**設立に関するシンポジウムに登壇する予定...
Mountain	14	44	**トルバチク山**噴火、カムチャツカ半島 - ナショナルジオグラフィック ニュース URL HASH
Park	11	30	本日、三重県伊勢市に「**伊勢フットボールヴィレッジ**」がオープン。新設のサッカー場の1面には...
Others (16 types)	66	251	...2015年6月開館予定の図書館をつくる計画が進んでいる。...を目指す「**男木島図書館**」URL
GROUP	**366**	**2173**	
Company	259	1441	「感情エンジン」とクラウドサービスを提供する企業 **Cocoro SB**を設立。状況と感情を経験として...
SoccerClub	55	304	東海社会人リーグ二部への昇格を決めたヴィアティン桑名が、チーム名を「**ヴィアティン三重**」に変更...
Organization	28	179	訂正。自民党の石破派は28日に約20人で旗揚げ。派閥の名称は「**水月会**」と決まった。 URL URL
PoliticalParty	24	249	時事ドットコム:滋賀知事が新党検討=脱原発掲げ「**日本未来の党**」【12衆院選】URL
OTHER	**130**	**561**	
Species	77	337	新種のカエル発見とニュースで。「**サドガエル**」だって。佐渡は固有種が多いって > URL
CelestialBody	17	75	最も地球に似た惑星?「**グリーゼ832c**」が新発見された...
MilitaryConflict	7	9	最新の**アレッポの戦い**の復元地図。やりたいことというかやろうとしてることというかやってることは明確で、
ChemicalCompound	7	16	...細胞膜ではなくメナキノンを標的とする新たな抗生物質・**ライソシンE**を発見。MRSAの新しい治療薬として...
Disease	1	8	社速: 新種ウイルスは**中東呼吸器症候群** WHOが命名: USER URL URL
Aircraft	1	5	ワイドボディ機史上最多の259機(LH34機,EY25機,QR50機,EK150機)の発注意向を得て**ボーイング777X**ローンチ
Others (4 types)	20	111	日立、英国高速鉄道向け新型車両「**クラス800シリーズ**」を公開。非電化区間も走行できるようモーター以外に...
UNMAPPED	7345	35552	気象庁、9月の豪雨を「**平成27年9月関東・東北豪雨**」と命名 URL SERIEの兄弟機!AQUOS Phone ZETA **SH-09D**が発表!: シャープらしからぬ洗練されたデザイン!あのエコ技も...
Total	19604	111046	

Table 3.3: Statistics of the Japanese emerging entities and their contexts obtained from our Twitter archive by our time-sensitive distant supervision.

3.5.1 Data

We adopt Twitter as a microblog and target English and Japanese, which are the top two languages on Twitter [8]. We use our archive of Twitter posts that are retrieved[3] by using the official Twitter APIs[4] and consists of more than 50B posts (32% are English and 20% are Japanese; This does not deviate much from the actual data [8]).

In Step 1 of § 3.4.1, we collected titles of articles that were registered in Wikipedia from March 11th, 2012 to December 31st, 2015, using the Wikipedia dump on June 20th, 2018. We then excluded redirects and disambiguation pages from the titles and then ran Step 2. We obtained a total of 398,706 English and 222,092 Japanese tweets, including the same number of emerging and prevalent contexts for 30,945 English entities and 19,604 Japanese entities as the training data, respectively. For model selection, we used 10% of the training data as the development data. We removed URLs, usernames, and hashtags from those texts.[5]

We then analyzed the obtained emerging contexts by mapping the included emerging entities to their corresponding fine-grained types assigned in the DBpedia ontology; for example, the entity "Spider-Man: Homecoming" is mapped to the type "Film." We further designed a coarse type for each language based on [22] and classified mapped types accordingly. We show the resulting training dataset in Table 3.2 and 3.3. The difference in the number of coarse types comes from the level of DBpedia maintenance for each language. Out of the 30,948 English and 19,604 Japanese emerging entities in our dataset, we have 19,120 (206 types) and 12,259 (51 types) type-mappings, respectively. For both English and Japanese, the entity types that are manually categorized into Person and Creative Work account for a large proportion. This is because these entities tend to generate a great deal of attention at the time of their appearance than other entities. It is interesting to note that the frequency of fine-grained types varies by language; for example, Person type of English includes many athlete type

[3]Starting from 26 popular Japanese users in Mar. 2011, their timelines (recent tweets) have been continuously collected using user_timeline API, while the user set has iteratively expanded to those who were mentioned or whose tweets were reposted by already targeted users.

[4]https://developer.twitter.com/en/docs/twitter-api

[5]For Japanese, we tokenized each example by using MeCab (ver. 0.996)[6] with ipadic dictionary (ver. 2.7.0).

entities at the top, but the Japanese ones do not. This indicates that the tendency for emerging entities to be added to Wikipedia varies from language to language. The UNMAPPED entities included disease, typhoon, and other terminology because there are no mappings for them in the DBpedia ontology. We also see that emerging contexts could be diverse according to the type of entity they include. We thus have to capture those contexts properly to discover various types of emerging entities.

As a further analysis, we apply the pattern mining algorithm PrefixSpan [31] to the positive and negative examples of each type of the training data to extract patterns that occur frequently in emerging contexts, and calculate the following score function using the frequency of each obtained pattern:

$$score(p) = \frac{PrefixSpan(p)_{positive}}{PrefixSpan(p)_{negative} + 1} \tag{3.1}$$

$PrefixSpan(p)_*$ is the frequency of the pattern p in the given examples. The score is higher when the pattern occurs more in the positive examples and less in the negative examples, *i.e.,* when it seems to be specific to the emerging contexts. Here, the minimum support value was set to 50, and patterns that contained symbols or only numbers were removed.

For both languages, we list the top-50 scored patterns for each coarse-type in Table 3.4 and 3.5. In both Japanese and English, we see that the obtained emerging contexts contain words that suggest the novelty of the emerging entity. Some of the words (*e.g.,* 'announce' and 'new' in English) appear in common across types. Therefore, it is important to capture the type-specific words and expressions when performing tasks like entity typing. In addition, in some of the types, words of a specific topic (*e.g.,* Baseball players' draft in PERSON type) appear concentratedly. This indicates that there is a bias in the fine-grained type of entity registered in Wikipedia.

3.5.2 Models

The following models were implemented for comparison:

TYPE	extracted patterns
PERSON	newest, acquire, first player, Rule, choice, model, Well done, undrafted, selects, enter, defeats, Atlanta, overall select, first ever, first pick, select overall, appointment, pick 2015, pick 2014, rushing, KO, becomes first, highest, threw, charge, AA, Bowl, new Miss, New Zealand, coach coach, congratulations, passed away, torn, Pakistani, 11th, press conference, chosen, TKO, executive, Proud, scouting, Pansare, unofficial, Congrats, loses, completed, Defensive, Conservative, Congratulations new, signed contract
CREATIVE WORK	Announced, Listen new, titled, announce new, new track, confirmed, new new, release new, new game, announces new, announce album, brand, Announce, brand new, announced new, premieres, first look, sequel, Announces New, drops, album called, first new, Watch trailer, proud, release album, Watch new, new coming, Watch video, announce new album, Announces Album, announces album, RPG, announcement, first album, First Look, reveal, Super 3D, New album, PS Vita, new series, album new, return, produced, Spring, new released, Hear new, Revealed, first trailer, Ubisoft, new film
LOCATION	name, Police, live, Zikim, us, attack, love, students, President, called, police, village, today., launch, reports, make, Army, stadium, home, Live, new, Hall, place, opening, Russian
GROUP	name, called, live, think, announce, campaign, free, Congrats, big, buy, raises, launches, data, PM, say, announced, news, startup, set, join, US, day, much, way, Thanks, play, named, Entrepreneurship, look, become, launched, group, Good, Dish, logo, Heinz, made, proud, One, PlayStation
EVENT	nominated, Best, Awards, new, Silva, set, first, host, wins, announced, today, vs, Silva UFC
DEVICE	announced, Motorola, display, X Style, HD, Play, Windows, Sony Z, official, Kindle, Kindle Fire, Nokia Nokia, Sony Xperia Z, camera, looks, unveiled, launched, Wear, Snapdragon, Sony Xperia Z3, Kindle Fire HDX, Android Wear, today, Lumia Lumia, look, Redmi Note, screen, Hands, Display, Aston Martin DB10, revealed, announces, Note Edge, Rs, specs, concept, coming, Samsung Galaxy Note Edge, Lumia Windows, Bond

Table 3.4: Extracted patterns of the English emerging contexts obtained from the training data.

TYPE	extracted patterns
PERSON	位 指名, 巡, 指名 選手, 指名 投手, ドラフト 位 指名, 選択, 出演 決定, お披露目, 選 当選, 市長 当選, 発表 さ れ, 市長 氏, 選手 内定, オーディション, デビュー 決定, し デビュー, ドラフト 会議, 内定, 選手 加入, ドラフト 中日, 中日 西武, ドラフト 投手 投手, 当選, 新人 し, メンバー し, ドラフト 選手, 選, ドラフト 広島, 位 中日, JR, 確定, デビュー し, 巡 中日, ドラフト 阪神, ドラフト 巡, ドラフト 楽天, 楽天 西武, 東日本, 阪神 楽天, 広島 阪神, ドラフト 巨人, 広島 西武, 中日 楽天, 位 阪神, ドラフト 西武, 中日 広島, 位 選手, 広島 楽天, 中日 阪神, 巡 楽天
CREATIVE WORK	シングル 決定, シングル 発売 決定, アルバム 決定, リリース 決定, 解禁, 公開 決定, 発売 初回 盤, ニュー 発売, シングル リリース, ニュー 決定, 番 組 スタート, アルバム 発売 決定, シングル 予約, 映画 決定, 発売 通常, 初回 通常 盤, 決定 発売, 盤 通常, ニュー シングル 発売, 監督 公開, タイト ル 決定, ニュー シングル 決定, 盤 通常 盤, 邦題, シングル タイトル, 発表 さ れ, 発売 限定 盤, 初回 盤 通常, アニメ 決定, アルバム リリース, 決定 予約, 発売 通常 盤, シングル 初回, ニュー 発売 決定, 発売 初回 通常, 初回 盤 通常 盤, 主題歌 決定, 発売 開始, 発売 盤 盤, 速報 決定, ドラマ 出 演, 出演 決定, アニメ テーマ, 決定 盤, オープン, 決定 初回, 発売 タイトル, スタート し, 決定 公開, 決定 曲
LOCATION	オープン, モール, 開業, オープン し, ニュース, スタバ, 鳥取, 日本, 施設, 予定, 発表, JR, スタバ 鳥取
GROUP	設立, 新党, 旭化成, 新党 党, 名称, 決定, 横浜, チーム, 事業, 会社 設立, マンション, 向け, 結い 党 党, 鉄道 鉄道, 結い 党 民権, 結い 党 民権 党, 日本 日本, 江田, 社名, データ, 歴史, TPP, 秋葉原 ラジオ ストアー 閉館, 所属, グループ, なり, 日本 原発 する, 秋葉原 64, 旭化成 データ, お知らせ, 発表 し, TPP する, 在来, 秋葉原 ラジオ ストアー 64, 秋葉原 64 歴史, 旭化成 建材 データ, 日本 TPP, 原発 する 党, 秋葉原 ラジオ ストアー 64 歴史, サッカー, 日本 TPP 原発, 日本 実現, 日本 原発 党, 減税, 減税 日本 党, 秋葉原 ラジオ ストアー 64 歴史 幕, 会社 鉄道, FC FC, 減税 日本 TPP, 会社 し

Table 3.5: Extracted patterns of the Japanese emerging contexts obtained from the training data.

Proposed (LSTM-CRF): We used the implementation of LSTM-CRF using Theano (ver. 0.9.0) provided by [41].[7] We set hyperparameters (Table 3.6) as suggested in [80], who explored the practical settings of neural sequence labeling. We optimized the model using stochastic gradient descent and chose the model at the epoch with the highest F_1 on the development data. To initialize the embedding layers, for English, we used pretrained 200-dimensional word embeddings using GloVe [61] [8] from 2 billion English tweets. For Japanese, we trained 200-dimensional word embeddings using GloVe from 800 million Japanese tweets posted from March 11th, 2011 to March 11th, 2012.

Baselines: Since our methods use automatically constructed training data, we prepared two baselines that do not utilize such data. Baseline1 regards NEs obtained by NER as emerging if they are not detected as NE on Twitter from one year to one week before the posting time of the input tweets. We set the period up to one week before to find NEs that emerge near the target day. Baseline2 regards NEs obtained by NER as emerging if they do not exist in a KB [54]. We regard the obtained NEs as emerging when they are not registered in Wikipedia as of the month before the posting time of the input tweets because there is a time lag to use the latest Wikipedia dump in actual settings. To make NER robust, we use LSTM-CRF trained with a dataset of WNUT17[22][9] for English, combined dataset of KWDLC[10] and KNBC[11] for Japanese, respectively using the parameters in Table 3.6. All of which are corpora in which NE tags are attached to noisy Web text.

3.5.3 Evaluation Procedures

To evaluate the proposed method, we designed two evaluation procedures for emerging entities discovered from Twitter.

Precision: To evaluate the precision of the obtained emerging entities, we applied each model to daily tweets, ranked the discovered entities using their confidence scores, and finally computed the accumulative precision for the top 500 entities.

[7] https://github.com/glample/tagger/
[8] glove.twitter.27B.zip from https://nlp.stanford.edu/projects/glove/
[9] https://noisy-text.github.io/2017/
[10] http://nlp.ist.i.kyoto-u.ac.jp/index.php?KWDLC
[11] http://nlp.ist.i.kyoto-u.ac.jp/kuntt/#ga739fe2

Parameter	Value
Character embedding size	30
Word embedding size	200
Dimension of Character Bi-LSTM	50
Dimension of Word Bi-LSTM	200
Batch size	32
Dropout	0.5
SGD learning rate	0.015

Table 3.6: Hyperparameters of LSTM-CRF.

As the test sets, for both English and Japanese, we randomly picked three sets of daily retweets, on March 21st, 2017 (2,699,377 English and 1,695,423 Japanese tweets), June 14th, 2017 (2,748,257 English and 2,041,833 Japanese tweets), and September 1st, 2017 (2,725,111 English and 1,901,305 Japanese tweets) so that the seasons do not overlap. As the confidence score of Proposed (LSTM-CRF), we used the marginal probability obtained using the constrained forward-backward algorithm [19]. We adopted the maximum scores for the extractions when several mentions of the same entity were recognized. Since Baselines do not provide any scores regarding the emergence of entities, we used the number of extractions of each entity normalized with the extraction number of the previous day as the confidence score. This captures the bursty feature that considers the appearance ratio of the previous day.

We asked three annotators, including the first author and two student volunteers, to decide whether the outputs were accompanied by emerging contexts defined in § 3.2 by referring to the input tweets and then adopt the majority labels to mediate the conflicts. We obtained an inter-rater agreement of 0.714 for English results and 0.798 for Japanese results, by Fleiss's Kappa [27], which indicates substantial agreement. These high agreements justify the solidness of our definition of the emerging entity and the task setting.

Relative Recall and Detection Immediacy: To evaluate the recall and detection immediacy of the obtained emerging entities, we ideally want to refer to the complete list of entities that have emerged in certain periods. However, it is unrealistic to have such a list for a diverse range of entities, including long-tail emerging entities. We instead evaluated the relative recall and immediacy against

a KB, by determining how many entities registered in Wikipedia could be found from the tweets and how early they were detected against their registration date in Wikipedia.

Since entities newly registered in Wikipedia include both emerging and prevalent entities, we obtained the reference list of emerging entities as follows. For both English and Japanese, we collected entities that appeared more than 100 times on our Twitter archive from January 1st, 2017 to June 20th, 2018, and then extracted retweets containing each entity since the first appearance. To exclude prevalent entities as much as possible, we ignored entities that appeared more than five times on our Twitter archive from March 11th, 2011 to March 11th, 2012. We obtained 15,811 English entities with 8,736,847 tweets (552 tweets per entity on average) and 13,406 Japanese entities with 9,108,612 tweets (679 tweets per entity on average) since March 12th, 2012, and then applied our method to these tweets and calculated the recall and detection immediacy of the obtained entities.

3.5.4 Results and Analysis

Precision and classification of detected emerging entities Figure 3.2 depict the cumulative precision (precision@k) for the top 500 entities discovered with each model for English and Japanese, respectively. Proposed (LSTM-CRF) is superior to the others and mostly maintained a precision above 70% on English and 80% on Japanese results (on average 73.4% and 83.2% for top-500 entities for the three sets of daily retweets, respectively), while two Baselines remained mostly under 30% on English and 20% on Japanese. The reason for the low precision of the Baselines is that these methods completely ignore emerging contexts and detect many prevalent entities and noises as a result. Although these prevalent entities can be removed to some extent by checking against KBs and past frequencies, it is difficult to deal with their spelling discrepancies. The poor precision of the proposed method in English compared to that in Japanese may be due to the noise in distant supervision. For example, unlike the Japanese data, the English one has the largest number of PERSON-type entities (Table 3.2). However, the novelty of this type of entity is ambiguous (it is unclear whether a person is emerging when he makes debut or when he becomes famous), and thus it introduces noisy examples into the collected contexts.

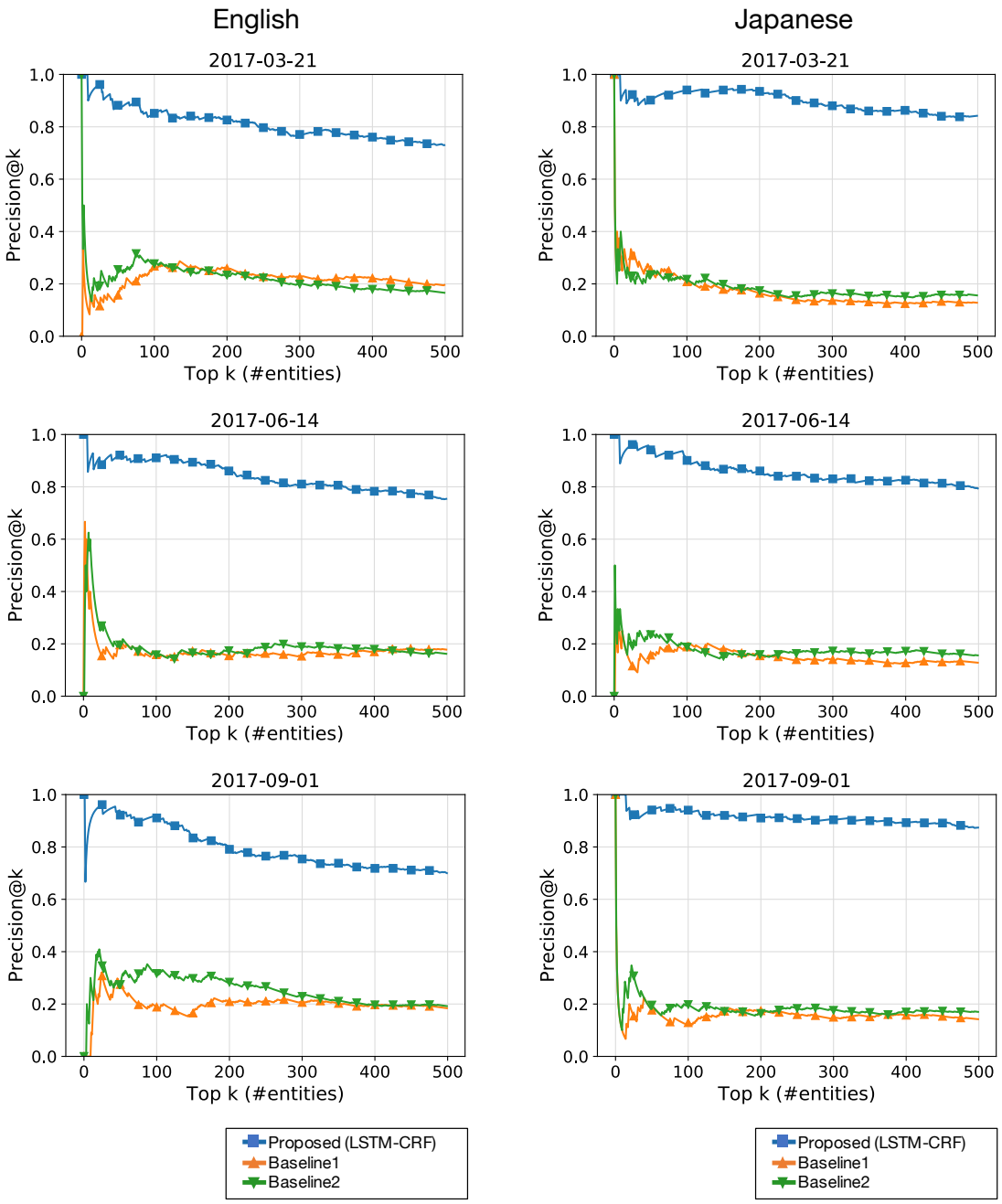

Figure 3.2: Precision@k for the top-500 emerging entities obtained from English and Japanese Twitter streams by each model.

daily tweets	HEAD (n > 100)	LONG-TAIL (n ≤ 100)	HOMOGRAPH	total
Mar. 21st, 2017	105	120	137	362
Jun. 14th, 2017	93	129	153	375
Sep. 1st, 2017	92	108	150	350

Table 3.7: Details of the English emerging entities discovered from the daily tweets with Proposed (LSTM-CRF).

daily tweets	HEAD (n > 100)	LONG-TAIL (n ≤ 100)	HOMOGRAPH	total
Mar. 21st, 2017	227	110	84	422
Jun. 14th, 2017	214	106	77	397
Sep. 1st, 2017	261	110	66	437

Table 3.8: Details of the Japanese emerging entities discovered from the daily tweets with Proposed (LSTM-CRF).

Table 3.7 and 3.8 list the detected emerging entities falling under three categories for both languages to confirm whether our method could discover various types of emerging entities defined in § 3.2. HEAD represents entities whose surfaces appeared over 100 times in our Twitter archive from the detection date to one year later, and LONG-TAIL is less than that. HOMOGRAPH represents homographic entities whose namings are already registered in Wikipedia before the detection date. As a result, Proposed (LSTM-CRF) could discover not only entities that would be added to Wikipedia but also find many long-tail emerging entities (*e.g.*, good and evil (play), Photo X Art Field (exhibition)) for both English and Japanese data. They are useful for companies performing social listening and local users trying to find something interesting even if their frequency is low. It also found homographic entities (*e.g.*, NEVER LAND (music album), Summer of Love (musical movie)), which were not found with the Baselines. Although [34, 25] reported that these homographic emerging entities are difficult to find, our method successfully discovered these entities by exploiting the emerging contexts of the entities. It is interesting that among the discovered emerging entities, HEAD is the most common in Japanese, while HOMOGRAPH is the most common in English data. This may be because the number of PERSON-type entities

that are likely to be homographic than others is the most frequent in the English training data (Table 3.2), and thus the model built using this data consequently detected many of such entities.

Relative recall and detection immediacy As the evaluation of relative recall, we focused on the best-performing method, *i.e.,* Proposed (LSTM-CRF) and computed its relative recall over the reference list of 15,811 English and 13,406 Japanese emerging entities. We detected 10,206 (64.5%) English and 10,852 (80.4%) Japanese emerging entities. These values are reasonably high considering that there was noise in the reference lists, such as a concept name that was defined after it became prevalent (*e.g.*, Virtual Youtuber) and periodic entities (*e.g.*, Tokyo prefectural election, 2017).

Table 3.9 and 3.10 show the distribution of the types of target entities for both languages obtained by the DBpedia mappings, detection ratio, and lead-time against the Wikipedia registration time for each type. As a result, our model found 10,206 (64.55%) entities from English and 10,852 (80.95%) entities from Japanese data. The cause of low detection rates in English may be due to the noise in the training data. For Creative Work type, our model achieved higher recall (79% for English and 89% for Japanese) compared to other types of entities for both languages. This suggests that this type of entity has less noise in the training data and is easier to detect due to its distinct emerging contexts. On the other hand, for the remaining types, detection rates varied according to the language. This may suggest that besides the effects of noise in the training data, the emerging nature of these entities varies to some extent across languages. Note that we found that some of those entities do not appear in emerging contexts at all within our Twitter archive. Since our method utilizes such emergence signals as the clue, it is difficult to discover entities appearing without emerging contexts. This is especially noticeable in English since English Wikipedia has many translations of articles (entities) in other languages. Many of those entities are not mentioned enough on English Twitter, and thus we cannot gather their emerging contexts. This is the current limitation of our method. Note that the referenced entities used in this evaluation included some noisy prevalent entities (*e.g.,* local company or minor person that are mere out-of-KB at the time of collecting the reference list) that might also affect the performances.

TYPE DBpedia types	# entities	# found (%)		lead-days mean	(median)
PERSON	**4968**	**3333**	**(67.09%)**	**896**	**(971)**
Person	1221	719	(58.89%)	781	(797)
AmericanFootballPlayer	638	432	(67.71%)	1046	(1155)
Politician	605	376	(62.15%)	671	(601)
SoccerPlayer	574	410	(71.42%)	984	(1093)
BasketballPlayer	343	250	(72.89%)	1186	(1272)
BaseballPlayer	219	148	(67.58%)	864	(895)
Cricketer	175	145	(82.86%)	1019	(1062)
IceHockeyPlayer	132	89	(67.42%)	973	(1000)
Others (63 types)	1056	764	(72.35%)	893	(962)
CREATIVE WORK	**2573**	**2045**	**(79.47%)**	**442**	**(187)**
Film	542	415	(76.57%)	479	(243)
TelevisionShow	452	363	(80.31%)	476	(153)
Album	423	372	(87.94%)	475	(182)
VideoGame	410	312	(76.10%)	338	(103)
Book	176	153	(86.93%)	512	(346)
Single	168	147	(87.50%)	408	(90)
Song	121	99	(81.82%)	323	(12)
Software	48	28	(58.33%)	346	(262)
Others (15 types)	233	156	(66.95%)	451	(247)
LOCATION	**645**	**347**	**(53.80%)**	**699**	**(687)**
City	207	45	(21.74%)	580	(425)
Building	102	80	(78.43%)	695	(725)
School	45	43	(95.56%)	946	(1096)
Stadium	33	24	(72.73%)	716	(708)
River	33	24	(72.73%)	897	(891)
Bay	22	18	(81.82%)	815	(920)
Others (33 types)	203	113	(55.67%)	703	(725)
GROUP	**556**	**360**	**(64.75%)**	**558**	**(430)**
Company	208	109	(53.44%)	535	(384)
Band	90	52	(57.78%)	737	(638)
Organisation	82	55	(67.07%)	426	(313)
MilitaryUnit	42	34	(80.95%)	572	(492)
PoliticalParty	34	33	(97.06%)	621	(448)
Others (14 types)	100	77	(77.00%)	459	(339)
EVENT	**178**	**142**	**(79.78%)**	**272**	**(19)**
Award	49	42	(85.71%)	219	(29)
WrestlingEvent	24	19	(79.17%)	566	(393)
SpaceMission	22	9	(40.91%)	-117	(-121)
MilitaryConflict	21	13	(61.90%)	856	(883)
MixedMartialArtsEvent	19	19	(100.00%)	-10	(-1)
Others (11 types)	43	40	(93.02%)	221	(48)
DEVICE	**157**	**90**	**(57.32%)**	**354**	**(131)**
Device	61	45	(73.77%)	294	(126)
Ship	44	16	(36.36%)	704	(590)
Automobile	25	12	(48.00%)	360	(184)
InformationAppliance	15	13	(86.67%)	110	(1)
Others (4 types)	12	4	(33.33%)	417	(298)
OTHER	**202**	**130**	**(64.36%)**	**474**	**(244)**
Owl:Thing	114	87	(76.32%)	506	(337)
Horse	39	25	(64.10%)	384	(240)
Others (16 types)	49	18	(36.73%)	447	(175)
UNMAPPED	6532	3759	(57.55%)	591	(456)
Total	15811	10206	(64.55%)	656	(570)

Table 3.9: Relative recall and time advantage over entity types of English emerging entities detected with Proposed (LSTM-CRF).

TYPE DBpedia types	# entities	# found (%)		lead-days mean	(median)
PERSON	**3851**	**3238**	**(84.08%)**	**660**	**(550)**
Actor	651	514	(78.96%)	742	(727)
MusicalArtist	624	463	(74.20%)	740	(649)
SoccerPlayer	477	429	(89.94%)	769	(759)
VoiceActor	345	323	(93.62%)	498	(363)
AdultActor	306	300	(98.04%)	376	(297)
BaseballPlayer	238	213	(89.50%)	591	(405)
Model	225	210	(93.33%)	764	(654)
Person	173	138	(79.77%)	777	(792)
Politician	136	112	(82.35%)	665	(538)
Wrestler	116	97	(83.62%)	360	(165)
Presenter	86	77	(89.53%)	635	(552)
Writer	83	65	(78.31%)	504	(376)
Others (14 types)	391	294	(75.19%)	768	(715)
CREATIVE WORK	**4122**	**3703**	**(89.84%)**	**377**	**(176)**
TelevisionShow	699	644	(92.13%)	225	(54)
MusicSingle	653	594	(90.96%)	304	(85)
Film	641	555	(86.58%)	388	(214)
MusicAlbum	550	499	(90.73%)	356	(161)
Manga	523	486	(92.93%)	672	(594)
VideoGame	440	392	(89.09%)	406	(251)
RadioProgram	228	203	(89.04%)	261	(38)
Anime	216	188	(87.04%)	254	(85)
Book	101	95	(94.06%)	621	(462)
Website	24	16	(66.67%)	869	(624)
Song	23	21	(91.30%)	549	(301)
Software	22	9	(40.91%)	817	(723)
Artwork	2	1	(50.00%)	160	(160)
LOCATION	**223**	**179**	**(80.27%)**	**597**	**(385)**
Building	89	73	(82.02%)	497	(291)
Museum	33	32	(96.97%)	685	(447)
Station	25	21	(84.00%)	264	(154)
School	18	9	(50.00%)	553	(62)
Library	13	13	(100.00%)	995	(1328)
Park	11	9	(81.82%)	639	(193)
University	7	6	(85.71%)	904	(996)
Temple	7	4	(57.14%)	1457	(1446)
Shrine	6	3	(50.00%)	528	(304)
ArchitecturalStructure	4	1	(25.00%)	912	(912)
Dam	3	1	(33.33%)	546	(546)
RailwayLine	2	2	(100.00%)	7	(7)
Others (5 types)	5	5	(100.00%)	1046	(999)
GROUP	**240**	**152**	**(63.33%)**	**545**	**(396)**
Company	188	116	(61.70%)	500	(359)
SoccerClub	26	13	(50.00%)	780	(741)
Organisation	16	14	(87.50%)	706	(416)
PoliticalParty	10	9	(90.00%)	552	(471)
OTHER	**59**	**18**	**(30.51%)**	**758**	**(977)**
Species	53	14	(26.42%)	825	(1008)
owl:Thing	8	5	(62.50%)	847	(736)
CelestialBody	3	1	(33.33%)	2	(2)
Train	2	2	(100.00%)	241	(241)
Aircraft	1	1	(100.00%)	1613	(1613)
UNMAPPED	4883	3557	(72.84%)	691	(615)
Total	13406	10852	(80.95%)	571	(406)

Table 3.10: Relative recall and time advantage over entity types of Japanese emerging entities detected with Proposed (LSTM-CRF).

We next evaluated detection immediacy. We found that 93.2% of the discovered English entities (9,511 out of 10,206) and 92.4% of the discovered Japanese entities (10,030 out of 10,852) were detected earlier than their registration in Wikipedia. We then investigated the remaining entities and found that there were periodic events such as Olympics and election or incorrectly included prevalent entities. The mean (and median) lead days of the first day when Proposed (LSTM-CRF) detected each entity against their registration date were 656 (and 570) for English and 571 (and 406) days for Japanese, which supports the detection immediacy of our method. One of the possible reasons English entities are detected earlier than Japanese ones is the effect of translated entities that are not easily mentioned in English Twitter. For both languages, compared to Creative Work types of entities, our method detected Person and Location types of entities earlier than their registration in Wikipedia, which means those entity types take longer to be notable enough to be registered in Wikipedia [30].

Overall, these results reconfirm that microblogs are useful sources for finding emerging entities and that our method can detect such entities at the early stage of their appearance. It also implies that relying on Wikipedia for the source of entities misses valuable information on emerging entities.

Examples Finally, we show the examples of predictions in the evaluation of precision with Proposed (LSTM-CRF) in Table 3.11. Even when the length of the post is short, and there are few clues as in the first example, or the target has homograph as in the second example, Proposed (LSTM-CRF) recognized entities correctly by utilizing emerging contexts.

We also found that some false positive predictions were caused by words often found in emerging contexts, such as 'new' and 'launch' in the third and fourth examples. It is preferable to use the global features of multiple posts for such examples rather than referring to only a single post to make predictions.

3.6 Chapter Summary

We introduced a novel task of discovering emerging entities in microblogs (§ 3.1, 3.2). We pointed out the problems of related tasks (§ 3.3), which simply detect

Entity: **Xiaomi Mi 6** True type: Device
Xiaomi Mi 6 could be the first Chinese flagship to be powered by Snapdragon 835 SoC » PhoneRadar.

Entity: **It's You** True type: Single
Im Seulong's new single album '**It's You**' will be released on Jun 20.

Entity: **Cameraman Joe** True type: NULL
Cameraman Joe is shown the new interactive app from. It's hoped it'll attract more younger and interesting...

Entity: **Don't Start a Business** True type: NULL
Don't Start a Business, Launch a Kickstarter

Table 3.11: Examples that Proposed (LSTM-CRF) predicted correctly (above two) and incorrectly (below two) (English)

entities that are not registered in the KB by treating them as emerging entities. To target only truly emerging entities, we proposed an effective method for discovering emerging entities in microblogs by exploiting the specific contexts of those entities using time-sensitive distant supervision (§ 3.4), which utilizes the Wikipedia entities and timestamps of microblogs. Experimental results demonstrated that our discovering method performed accurately and showed that emerging entities, including homographic and long-tail ones, can be effectively and instantly discovered by obtaining emerging contexts (§ 3.5).

Chapter 4

Discovery of Disappearing Entity

4.1 Introduction

We always catch up with information about entities such as people, works, products, and events that appear in the real world and utilize them to make decisions in our daily lives. One of the most important pieces of knowledge for these entities is about their disappearance. For example, keeping track of people who have died or companies that have gone bankrupt is necessary to expand a knowledge base (KB) that accumulates knowledge about entities. People can prevent loss of opportunities by catching up information about events that will be discontinued, facilities and stores that will be closed, and products that will no longer be in service as soon as possible.

These *disappearing entities* can be captured by preparing a list of entities, such as KBs, and monitoring their appearances in a text. However, disappearing entities tend to keep getting mentioned even after they actually disappeared from the world, and conversely, they are no longer mentioned does not necessarily mean that they have disappeared. Therefore, it is difficult to determine what makes an entity disappear.

Another solution is to use entity linking [69], a technique for linking entities in the given text to the corresponding entries of KBs, to extract information from the linked entry whether the entity is disappeared or not. However, the update of the KBs is basically slow, and we cannot extract the necessary knowledge of disappearance when we need it. This becomes especially problematic when

detecting entities scheduled to disappear, such as events or services. Moreover, we cannot handle out-of-KB and long-tail entities because their entries do not exist in the KBs and consequently overlook these entities completely. Although other information extraction techniques such as named entity recognition (NER) and event extraction could be used, it is difficult to apply them as-is because there are no datasets specific to disappearing entities.

Given those situations, we take on the new task of discovering disappearing entities that appear in microblogs, where news and personal experiences are massively shared. We focus on the fact that when people mention disappearing entities in microblogs, they use specific expressions that suggest their disappearance. By properly capturing these contexts, we can discover and classify a variety of disappearing entities at the early stage of their disappearance. We use time-sensitive distant supervision (§ 3)[4], which collects entities' specific timing of contexts by utilizing Wikipedia entities and timestamps of microblogs to collect disappearing contexts and build a dataset. At this time, to collect the contexts of disappearing entities more accurately, we extract the year of disappearance for each entity described in Wikipedia and incorporate it into the distant supervision.

We train a NER model on the collected entities and contexts to discover disappearing entities. Here, we focus on the fact that when the entity disappears in the microblog, multiple posts mentioning that entity often appear and propose utilizing those posts to refine pretrained word embeddings and incorporate them into the NER model. This allows the model to attend to the tokens and expressions that frequently appear among multiple posts and to recognize disappearing entities robustly.

We applied the proposed method to both the English and Japanese datasets constructed from Twitter to evaluate the detection of disappearing entities. Experimental results confirmed that the proposed method overwhelmed the performance of a baseline, which collected the latest burst of posts about the disappearing entities as the disappearing contexts using time-sensitive distant supervision and used them for training a NER model. Moreover, as the evaluation of relative recall, our method successfully found more than half of the target disappearing entities in Wikipedia, and some of them were discovered on average one month earlier than the update of the disappearance in Wikipedia.

Our contributions are as follows:

- We introduce a novel task of discovering disappearing entities in microblogs.
- We modify the time-sensitive distant supervision method for collecting disappearing entities and contexts more accurately.
- We propose novel embedding features of NER, which capture multiple contexts of entities in the microblog
- Our method found disappearing entities accurately than a baseline and quickly than the updates of disappearance in Wikipedia.
- We will release the code of our model and all the datasets (tweet IDs) used in the experiments to promote reproducibility.

4.2 Definition of Disappearing Entity

In this section, we define what is meant by the term *disappearing entity* in this study. In § 3[4], we reported that emerging entities newly appearing in the real world have a process from their first appearance to the time when they become known to the public, and in this process they are referred to with specific expressions, *i.e.*, emerging contexts. Similar to this, we define disappearing entities and contexts by focusing on the fact that specific expressions indicating plans and signs of the disappearance appear in the contexts not only at the time of disappearance but also in the process up to that time as follows:

Disappearing contexts. *Contexts in which the writers assumed the readers do not know the disappearance of the entities.*

Disappearing entities. *Entities in the state of being still observed in disappearing contexts.*

By properly identifying these disappearing contexts, we can detect corresponding disappearing entities in their early stages. As shown in the examples in Table 4.1, some of the disappearing contexts contain preliminary notices of the disappearance. Therefore, we can even find disappearing entities before they actually disappear, such as events that are ending or products that are no longer supported, before they actually disappear. We later confirm the solidness of

I'm so *sad to hear* that **Dave Laing** has *died*. Dave was a very accomplished music industry journalist. But also made a mass . . .
Legendary Emmy Award - winning and former NBC4 anchor, **Doug Adair**, *passed away* peacefully alongside family on Monday in Pleasanton
ESPN is *shuttering* **ESPN Deportes Radio** this fall in what appears to be yet another cost cutting move.
RIP **Ed Corne** What a Physique for under 200lbs! This guy knew how to pose If schwarzenegger was impressed, you. . .
Nike *shuts down* Oregan Project ! Less than two weeks after the USADA handed a four - year *ban* to **Nike Oregon Project** coach
Family Circle, a pillar of women's magazines, will *shut down* after 87 years.
Can't believe **Google+** is being *shut down*. It's like when they just pulled Google Friends Connect all over again . . .
Green Mountain College *Announces Plan To Close* This Spring, Court Rules State Has To Refund Fee For Swanton Wind Pr . . .
Bernie Sanders on the *planned closure* of **Hahnemann University Hospital** in Philadelphia: "In the midst of a healthcare crisis . . .
RT : Here's your *Demolition* Day Planner for **Martin Tower**. A brief, stray shower can't be ruled out before the big BOOM! The . . .
The **Newseum** will *close* at the end of 2019 following the sale of its building to. Organization says . . .
Red Bull Air Race World Championship *will not continue* after 2019. URL
We had a blast **BronyCon** and we're *sad* that it's the *last one* for them. We hope all their staff have a great future . . .
Pristin to *disband* after 2 years promoting as a group + K - Netz express how raged they are towards Pledis Ent.

Table 4.1: Example tweets on disappearing entities (bold) with expressions suggesting their disappearance (italic).

these definitions by evaluating the inter-rater agreement of disappearing entities acquired from microblogs (§ 4.5.4).

4.3 Related Work

To the best of our knowledge, there has been no study attempting to find disappearing entities in the meaning of our study. We briefly review the current studies related to our task.

4.3.1 Entity Linking and Named Entity Recognition

Entity linking [69, 39, 49] performs the linking of entity mentions to corresponding KB entries. Although this allows us to obtain information about the disappearance of the entity from the linked entry, in many cases, the knowledge is often not updated at the time of the disappearance of the entity is confirmed, and the entry itself does not exist for long-tail or out-of-KB entities.

Named entity recognition (NER) [53, 41, 6, 5] performs the recognition and typing of entity mentions in a given text. Since the usual datasets for training NER models are specialized for just recognizing mentions [10], it is necessary to build dedicated training data for tasks that require context understanding along with entity recognition, such as discovering disappearing entities.

In this study, although we use NER to recognize entities, we devise a way to make the training data so that the model can properly understand the contexts.

4.3.2 Event Extraction and Temporal Slot Filling

As part of information extraction, some studies [66, 56, 46, 45] have tackled the task of extracting an event (*e.g.*, birth of a person) and its predefined attributes and arguments from the text. Although it is possible to use this technique to detect disappearing entities, many entity disappearances are not defined in well-known datasets such as ACE [23] and ERE [1], and is not suitable for detecting various disappearing entities.

KBP2011 [36, 50] introduced the task of identifying the duration of an event given text, entities (*e.g.*, Steve Jobs) and their events (*e.g.*, become CEO) called temporal slot filling. The events are attributes defined in Freebase, and they include some disappearance of entities such as a person's lifespan. However, the types of disappearance handled in this task are limited. Moreover, since the

entity mentions are given in advance for this task, it is difficult to handle entities that do not exist in the dictionary of entities such as KBs.

4.3.3 Emerging Entity Discovery

The counterpart to disappearing entities, we tried to find emerging entities that are newly born in the world (§ 3)[4]. To discover only truly emerging entities, we focused on the fact that people use expressions that suggest novelty when mentioning emerging entities and defined them based on these expressions (contexts). We also proposed a distant supervision method called time-sensitive distant supervision to collect those emerging contexts efficiently using KB entities and microblog timestamps and developed a NER model to detect emerging entities using the collected contexts.

We modify time-sensitive distant supervision for disappearing entities and utilize collected contexts to develop a NER model.

4.4 Proposed method

The proposed method discovers disappearing entities in microblogs. Here, we target Twitter, where various sources, including news articles and personal posts, are shared. We use time-sensitive distant supervision (§ 3)[4] that exploits Wikipedia entities and timestamped Twitter posts to construct the dataset. Here, to accurately collect disappearing contexts of disappearing entities (§ 4.2), we modify time-sensitive distant supervision to capture the timing of entity disappearance. To ensure that the model can make robust predictions with the constructed dataset, we refine pretrained word embeddings to acquire features about multiple occurrences of disappearing entities and feed them into the model.

4.4.1 Modified Time-sensitive Distant-supervision

To construct the dataset of disappearing entities, we first extract entities with unique namings that have already disappeared from Wikipedia and collect their disappearing contexts from the time-series Twitter posts using the method called time-sensitive distant supervision. However, unlike the emerging entities,

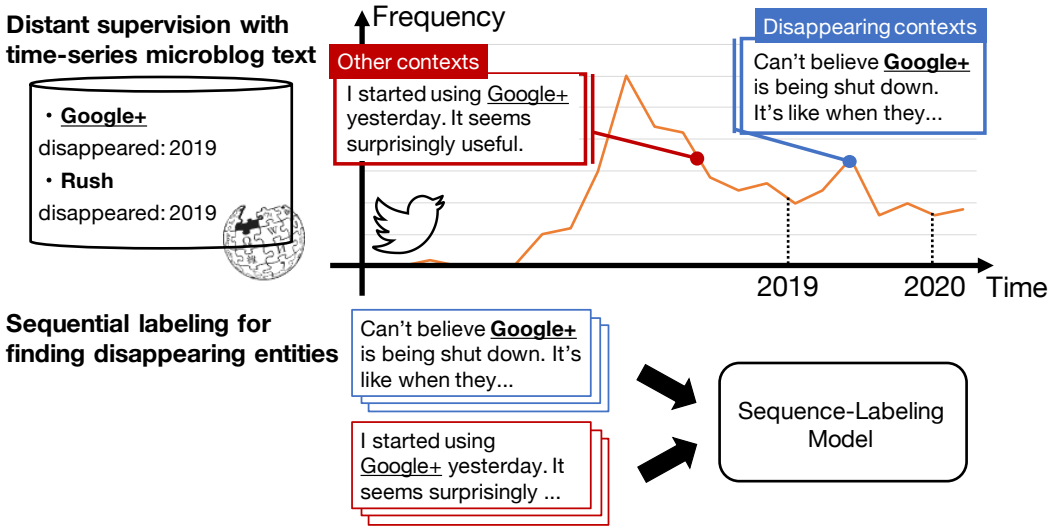

Figure 4.1: Time-sensitive distant supervision: for the entities retrieved from a KB, disappearing and other contexts are collected from microblogs by utilizing the year of entity disappearance, and a sequence labeling model is trained from the obtained contexts.

the number of disappearing entities is substantially small, and the timing of their disappearance is difficult to capture. Therefore, we explicitly feed the timing of entity disappearance extracted from Wikipedia to time-sensitive distant supervision and collect disappearing entities and their contexts more accurately (Figure 4.1). The specific procedure is as follows:

Step 1 (Collecting candidates of disappearing entities)

We start by collecting titles of articles in Wikipedia as entities. To ensure that we collect only entities that have actually disappeared, we refer to the list of disestablished entities in Wikipedia[1] and collect the titles of articles and their year of disappearance. We next excluded entities whose year of the first appearance on Twitter was the same as the year of their disappearance since they could be emerging entities and could contaminate the contexts. We also remove entities that have the ambiguity page so that the contexts are not contaminated by

[1]We collect the entities under the categories contained in the following pages. https://en.wikipedia.org/wiki/Category:Deaths_by_year and https://en.wikipedia.org/w/index.php?title=Category:Disestablishments_by_year

homographic entities, which share the same namings with other entities (*e.g.*, "Go" refers to a programming language, a board game, and a verb).

Step 2 (Collecting contexts of disappearing entities)

Compared to emerging entities, where the first appearance of the entity is often the emerging context, capturing the timing of entity disappearance is difficult since they keep getting mentioned in microblogs even after they actually disappeared. Therefore, for each collected entity, we utilize the year of disappearance described in Wikipedia and frequency of appearance in Twitter to gather their disappearing contexts. Specifically, we collect the last k posts of the day with the highest number of occurrences in the given year, assuming that the timing that received the most attention in the year of the disappearance includes the disappearing contexts.

In § 3[4], for each collected entity, we collected contexts that differed from the positive examples as negative examples to avoid that the NER model overfits to detect the mention of positive examples. We thus similarly collected random k non-disappearing contexts as negative examples from posts prior to the year in which we collected positive examples for each entity. This enables the model to discriminate between disappearing contexts and other contexts and reduces the effect of noisy contexts wrongly included in the positive examples.

We finally label only the acquired entities in the positive examples and combine them with their negative examples to form the training data for a NER model with sequence labeling. We set n to 100 as in § 3[4].

4.4.2 Finding Disappearing Entities

We train a NER model for finding disappearing entities from the collected training data. Unlike the emerging entities, it is difficult to robustly train the model since the number of disappearing entities is small. Moreover, short and noisy microblog posts deteriorate the performance of the NER model. We thus focus on the fact that when entities disappear in microblogs, multiple posts mentioning the disappearance of entities often appear. By obtaining features from these multiple posts, we can make the training and prediction of the model more robust, even from noisy microblog posts. To do this, we propose the unsupervised method of

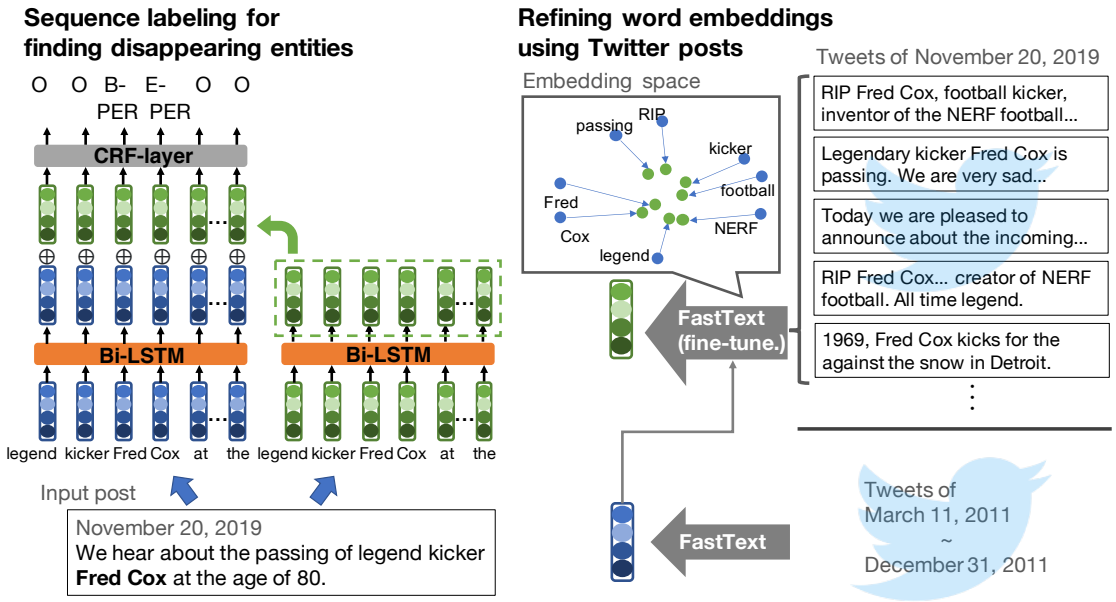

Figure 4.2: Sequence labeling with refined word embeddings: we fine-tune pretrained word embeddings using the Twitter stream on the day of the input post, and feed them into the LSTM-CRF model for robust training and prediction.

refining pretrained word embeddings as features from multiple posts of a Twitter stream and the sequence labeling model for finding disappearing entities using the refined word embeddings as additional input (Figure 4.2).

Refining Pretrained Word Embeddings

We want to extract features from multiple posts on the Twitter stream. However, since the target entity is unknown at the test time of NER, it is difficult to collect only relevant posts from the massive Twitter stream. Here, we note that many neural models input pretrained word embeddings and propose to fine-tune them using the posts on the day of detecting disappearing entities. This allows the refined word embeddings to reflect the tokens and their co-occurrences in the Twitter stream of the target day without the need for post selection as shown in Figure 4.2. The specific procedure is as follows:

First, we train the base word embeddings v_{base} using the posts prior to the period in which we collected the data in § 4.4.1. As the method of constructing word embeddings, we use fastText [12] to deal with unknown words.

Next, we use the Twitter stream at each date d of the posts collected in § 4.4.1 to fine-tune the word vector v_{base} for obtaining $v_{d}.Resulting\mathrm{v}_{d}$ can be interpreted as capturing the temporary semantic change of the word embedding v_{base} at date d, and it can be treated as an auxiliary input of models for various tasks.

Sequence Labeling with Refined Word Embeddings

We follow the method of § 3[4] and use sequence labeling to discover entities. We adopted long short-term memory with conditional random field (LSTM-CRF) [35] and flair embeddings [6] as the sequence labeling model. This model inputs pretrained word embeddings and character embeddings, which are encoded by the pretrained character-based bidirectional-LSTM language model, into the word-based bidirectional-LSTM and makes predictions through the CRF layer.

Based on this model, we prepare another word-based bidirectional-LSTM and input the refined word embeddings v_d corresponding to the date of the input post as shown in Figure 4.2. We then concatenate the hidden layers of each LSTM at each time and feed them into the CRF layer. This enables the model to consider the global information of the given Twitter stream other than the input post.

We adopted BIOES as the tagging scheme, which was reported to be better than other schemes [63]. We tagged disappearing entities in the positive examples with BIES and their type and the others with O.

4.5 Experiments

We applied the proposed method to the actual Twitter archive and performed our task of discovering disappearing entities.

4.5.1 Data

We constructed training, development, and test data for our task by using time-sensitive distant supervision detailed in § 4.4.1. We adopt Twitter as a microblog and target English and Japanese, which are the top two languages on Twitter [8].

TYPE Wikipedia categories	# entities	# posts	examples of disappearing context (truncated)
PERSON	**780**	**24689**	
Deaths	780	24689	**Roger Ailes** died of complications of a subdural hematoma after he fell at home, hit his head.
			. . . to ski one day with Olympic Legend **Stein Eriksen** at Deer Valley. He passed yesterday at. . .
			Billy Brewer has passed away. A wonderful guy who loved Coaching & was a personable guy. . .
			RIP **Dave Rosenfield**: @USER executive, International League schedule maker, one-time. . . URL
			The passing of Dr. **Irving Moskowitz** is a tremendous loss for the world, the Jewish people. . .
CREATIVE WORK	**975**	**24222**	
American_television_series	381	12413	RT @USER: See what's coming up in the **Diggnation** Finale airing next week. . .
British_television_series	139	3650	Wait. . . is this episode supposed to be the **Metalocalypse** finale or the start. . .
Korean_television_series	75	2039	Late Friday news dump: Comedy Central has canceled "**Not Safe with Nikki Glase**". . .
Web_series	57	1679	Final Series of **McLevy** this week on @USER The Scotland office are missing cast & crew . . .
Philippine_television_series	56	875	. . . didnt see it coming but still frustrated as all out that **Tekzilla** is now cancelled. . .
Canadian_television_series	44	570	Last Episode of **Star Trek Continues** was released today "To Boldly Go: Part II". . .
Others (33 types)	223	2996	Crying while watching the last episode of **Packed to the Rafters** :(such a great tv show
LOCATION	**240**	**3545**	
Buildings_and_structures	54	949	@USER: After 67 years **Clemson House** is gone in seconds. (This version is sped up) URL
Educational_institutions	36	647	**Coleman University** closing its doors after loss of accreditation URL
Sports_venues	34	598	**Baylor's Floyd Casey Stadium** is no more. May you and your tarp rest in peace. . . URL
Restaurants	24	286	. . . food news of HASH Everyone loves Lynn's. **Lynn's Paradise Cafe** abruptly serves its final meal. . .
Populated_places	23	232	BREAKING: Macy's will close its **Landmark Mall** store as part of its swath of 2017 closures:
Others (18 types)	69	833	Sad news, crime sleuths. The **National Museum of Crime and Punishment** in D.C. will close at the. . .
GROUP	**1187**	**30384**	
Musical_groups	453	11559	. . . in Flyleaf anymore. Adam's not in Three Days Grace anymore. **My Chemical Romance** broke up. . .
Retail_companies	79	3187	The convenience store chain, **My Local**, is to be placed in administration,9 months after it was sold. . .
Airlines	60	1535	The second hand car giant **Carcraft** has gone bust, with the loss of around 500 jobs across the UK. . .
Political_parties	56	654	The **Australian Democrats** officially deregistered by AEC bc less than 500 members. - ABC news. . .
Mass_media_companies	44	1351	**Mad Catz** files for bankruptcy and is ceasing operations URL URL
Others (66 types)	495	12098	**The Foreign Policy Initiative**, a right-leaning foreign-policy think tank, will cease operations. . .
EVENT	**186**	**4940**	
Sporting_events	111	2492	The **Adidas Grand Prix** in NY, replaced by Rabat meet in Diamond League series, is set to become. . .
Events	33	1462	Live music blow. MT @TomTilley: C3 confirms **Big Day Out** will NOT go ahead in 2015.
Awards	20	646	**World Series Formula V8 3.5** to end after 2017 season URL URL
Sports_leagues	16	137	The cancellation of **Women's Professional Soccer League** in US is bad news for England & Team. . .
Music_festivals	6	203	**Camden Crawl** organisers issue full statement following liquidation, blaming poor ticket sales:. . .
SERVICE&PRODUCT	**709**	**17574**	
Magazines	187	2998	was what inspired me to write for mags. RT @USER: Future closes **Nintendo Gamer magazine**
Internet_properties	131	5916	Google is going to shut down **Orkut** on September 30. . . .Haven't seen that site since 2007 I. . .
Products_and_services	119	3450	**The PlayStation 3** has ended production according to the official PlayStation Japan website.
Television_stations	105	1727	If you get past the exciting times and digital strategy, the closure of **STV2** is four paragraphs. . .
Publications	53	1043	Joe weighs in on the demise of **The Tampa Tribune** and the changing Bucs beat. (full story) URL
Others (8 types)	114	2440	Just heard that **Yorkshire Radio** is no more. Great shame as @USER was pretty much the voice of . . .
TOTAL	**3213**	**81850**	

Table 4.2: Statistics of the English disappearing entities and their contexts in the training data obtained from our Twitter archive by using time-sensitive distant supervision.

TYPE Wikipedia categories	# entities	# posts	examples of disappearing context (truncated)
PERSON	**416**	**28049**	
没	416	28049	アンパンマンの**やなせたかし**さんがお亡くなりになりました。94歳でした。 本当にたくさんの夢と勇気を...
			改めて、ご冥福をお祈りいたします。 僕の「師」はこれからも**松原正樹**です。
			ひびきわたるの訃報。 演芸番組ではおなじみだった。爆笑を取るタイプではない「正統派の色物」であった...
			アンドレ・プレヴィン 逝く。奇しくも『8K版 マイ・フェア・レディ』が追悼上映になる。本作はジュリー...
			メイセイオペラを育てた**佐々木修一**調教師が死去(サンケイスポーツ) - Yahoo!ニュース URL
			山口勝弘氏が亡くなった。石山さんが設計した淡路の山勝工房にも行ったし、メディア・アートのパイオニア...
			...がこれほど似合う人もおるまいて。合掌。→マーベル・コミックの生みの親、**スタン・リー**氏が死去...
LOCATION	**341**	**9434**	
施設	138	4637	**TOHOシネマズ有楽座**が閉館 58年の歴史に幕(写真 全2枚) HASH HASH URL
鉄道駅	81	1614	RT @USER: 一日の利用者は 女子高生ただ一人 高校卒業と同時に 廃止になる駅 北海道 **上白滝駅** URL
建築物	60	1616	南町田駅周辺の再開発のアオリで今日閉館した**グランベリーモール**、流石に最後は凄い人だらけだった。
博物館	34	893	**名古屋ボストン美術館**来たけど今日で閉館ということですごい人数なのでやめとこう。普段からこのぐらい...
スポーツ施設	27	673	お疲れさまです。**横浜みなとみらいスポーツパーク**11月30日をもちまして運営を終了することとなりました...
道路施設	1	1	...5/20 22:00をもって運用終了(廃止)となった、首都高高速1号羽田線上り「**平和島本線料金所**」最終日の様子...
GROUP	**723**	**22404**	
企業	291	6064	**スタジオコクピット**解散。最初の5年ほどしか在籍してなかったけとれど、ほんとうに色んな勉強をせてもらった...
音楽グループ	271	13177	『**Aqua Timez**』、2018年の活動をもって解散へ URL
政府機関	45	1039	かつて在籍していた**検査局**が消滅しますが、余り感慨はなく、ただ、今後の監督行政が適切に行われていくことを...
教育機関	33	118	母校・**日活芸術学院**が閉校なので、閉校の集いと言うイベントに行ってきた!日活撮影所のステージを...
政党・政治団体	30	987	民主党と維新の党は、民主党の党名を変更した上で**維新の党**が解散し、民主党に合流することで大筋合意しました...
スポーツチーム	29	793	...滋賀・高島クが資金不足などで廃部へ 12年に日本選手権出場: 社会人野球の滋賀・**高島ベースボールクラブ**が...
組織	24	226	八木アンテナ発明に貢献 **斎藤報恩会**が解散=河北新報 URL 当時画期的だった民間による研究費助成に取り組...
EVENT	**90**	**3585**	
イベント	56	2469	【京都の宇治川花火 再開を断念】今夏まで4年連続で中止となっている「**宇治川花火大会**」について...
			タナバタビアフェスタトヤマ。1回目から欠かさず参加して来ました。最後になるのは残念だけど、全国の...
			紙面【社会】最後の**大江健三郎賞**に岩城 けいさん「さようなら、オレンジ」ほか詳しくは本日(4月7日付)...
スポーツイベント	34	1116	鈴鹿1000kmは沢山思い出があります。 良い事も悪いも。 今回で最後となる**鈴鹿1000kmレース** ...
			九州一周駅伝が今年の大会を最後に終了することが、正式発表されたそうです。確かな筋からの情報なので...
SERVICE&PRODUCT	**336**	**11630**	
雑誌	157	6986	既報の通り、**月刊IKKI**は9月25日発売11月号をもって休刊いたします。支えてくださった皆様、本当にありがとう...
			電撃ホビーマガジンって最終号なんだ!!そしてこのタイミングでサイバーフォーミュラ特集!(劇中の1年目が...%
オンラインゲーム	120	2843	【悲報】スマホゲー『**東京ファントム**』がサービス終了 iOS版リリースからたった2ヶ月後の出来事・・・
			『H1Z1』から改名したゾンビサバイバルゲーム『**Just Survive**』が10月サービス終了へ。早期アクセスを...
ウェブサイト	59	1801	ひとつの時代の終わりーファイル共有サービスの老舗、**RapidShare**が来月で閉鎖 URL URL
			な、な、なんと!! **漫画村** 閉鎖!!! 今後暇な時間どうしたらいいんだ〜って思ってるそこのアナタ! ここだけの話...
TOTAL	**1906**	**75102**	

Table 4.3: Statistics of the Japanese disappearing entities and their contexts in the training data obtained from our Twitter archive by using time-sensitive distant supervision.

TYPE **Wikipedia categories**	**#ent.**	**#posts**
PERSON	**147**	**422**
Deaths	147	422
CREATIVE WORK	**10**	**23**
American_television_series	6	16
Radio_programme	2	3
Web_series	1	2
Philippine_television_series	1	2
LOCATION	**46**	**113**
Sports_venues	7	19
Shopping_malls	5	15
Restaurants	5	10
Railway_lines	2	5
Populated_places	2	2
Others (8 types)	25	62
GROUP	**103**	**270**
Retail_companies	12	33
Video_game_companies	4	12
Transport_companies	4	11
Telecommunications_companies	1	2
Religious_organizations	1	3
Others (25 types)	81	209
EVENT	**8**	**19**
Recurring_sporting_events	4	7
Sports_leagues	3	9
Recurring_events	1	3
SERVICE&PRODUCT	**43**	**114**
Television_stations	5	12
Publications	4	10
Radio_stations	2	4
YouTube_channels	1	3
Space_probes	1	3
Others (6 types)	30	82
TOTAL	**357**	**961**

(a) English data

TYPE **Wikipedia categories**	**#ent.**	**#posts**
PERSON	**73**	**220**
没	73	220
LOCATION	**42**	**114**
施設	23	64
鉄道駅	10	26
建築物	7	18
博物館	1	3
スポーツ施設	1	3
GROUP	**64**	**173**
音楽グループ	29	84
企業	26	68
組織	7	17
スポーツチーム	2	4
EVENT	**9**	**25**
イベント	5	14
スポーツイベント	4	11
SERVICE&PRODUCT	**47**	**131**
オンラインゲーム	31	89
雑誌	11	29
ウェブサイト	5	13
TOTAL	**235**	**663**

(b) Japanese data

Table 4.4: Statistics of the English and Japanese disappearing entities and disappearing contexts in the test data obtained from our Twitter archive.

We use our archive of Twitter posts that are retrieved[2] by using the official Twitter APIs[3] and consists of more than 50B posts (32% are English and 20% are Japanese; This does not deviate much from the actual data [8]).

In Step 1 of § 4.4.1, we collected article titles of disestablished entities in Wikipedia from 2012 to 2019, using the Wikipedia dump on June 20th, 2021. To acquire entity types, we manually map the category in the page of the disestablished entity to the coarse-grained type;[4] for example, the entity 'Aqua Timez' is mapped to the type 'GROUP.' Here, since the number of entities for the type PERSON and CREATIVE WORK is much larger than the other types, we undersampled to 1000 entities for those types. We then performed excluding entities as described and then ran Step 2. From the collected data, we split the data up to 2018 for training data and 2019 for test data. As the training data, we obtained a total of 163,700 English and 150,204 Japanese tweets, including the same number of disappearing and other contexts for 3,213 English entities and 1,906 Japanese entities, respectively. For model selection, we used 10% of the training data as the development data. We removed URLs, usernames, and hashtags from those text[5]

As the test data from the collected disappearing entities of 2019, we randomly selected three posts for each entity in Japanese and English and asked three annotators, including the first author and two graduate students, to judge whether each of the contexts is accompanied by the disappearing context. We adopt the positive context with the answers agreed upon by two or more annotators. Using the negative examples of the collected entities, we asked the annotators to determine the non-disappearing contexts using the same procedure and collected the same number of positive examples. We obtained an inter-rater agreement of 0.722 for English and 0.786 for Japanese by Fleiss's Kappa; both show substantial

[2]Starting from 26 popular Japanese users in Mar. 2011, their timelines (recent tweets) have been continuously collected using user_timeline API, while the user set has iteratively expanded to those who were mentioned or whose tweets were reposted by already targeted users.

[3]https://developer.twitter.com/en/docs/twitter-api

[4]Although we used DBpedia mappings to assign the fine-grained type of emerging entities (§ 3)[4], since many of the mappings for disappearing entities were not defined, we used this method to obtain coarse-grained types.

[5]For Japanese, we tokenized each example by using MeCab (ver. 0.996).[6] with ipadic dictionary (ver. 2.7.0).

agreement. As a result, we obtained a total of 961 English and 663 Japanese tweets for 357 English entities and 235 Japanese entities as the test data, respectively.

We then analyzed the obtained disappearing entities and their contexts. We show the resulting training and test data in Table 4.2, 4.3 and 4.4. The difference between the coarse and fine-grained types comes from the level of Wikipedia compilation for each language. For both English and Japanese, the entity types that are manually categorized into PERSON and GROUP account for a large proportion. These types of entities, occupied by persons, musical groups, and companies, are more likely to disappear. We also see that disappearing contexts could be diverse according to the type of entity they include. We thus have to capture those contexts properly to discover various types of discovering entities.

As a further analysis, we applied the pattern mining algorithm PrefixSpan [31] to the positive and negative examples of each type of the training data to extract patterns that occur frequently in disappearing contexts, and calculate the following score function using the frequency of each obtained pattern:

$$score(p) = \frac{PrefixSpan(p)_{positive}}{PrefixSpan(p)_{negative} + 1} \tag{4.1}$$

$PrefixSpan(p)_*$ is the frequency of the pattern p in the given examples. The score is higher when the pattern occurs more in the positive examples and less in the negative examples, *i.e.*, when it seems to be specific to the disappearing contexts. Here, the minimum support value was set to 50, and patterns that contained symbols or only numbers were removed.

For both languages, we list the top-50 scored patterns for each coarse-type in Table 4.5 and 4.6. In both Japanese and English, we see that the obtained disappearing contexts contain words suggesting the disappearance of the disappearing entity. Some of the words (*e.g.*, 'RIP', 'sad' in English) appear in common across types. Therefore, it is important to capture the type-specific words and expressions when performing tasks like entity typing. In addition, for the CREATIVE WORK and EVENT types in English, there are few words that indicate the disappearance of the disappearing entity. This may be because there are few patterns of disappearance for these types.

TYPE	extracted patterns
Person	condolences, passes away, Sad hear, hear death, sad news, sad passing, away age, saddened hear, passed age, dies via, saddened passing, sorry, dies age, passed away age, Rest Peace, Former dies, died aged, deeply, Saddened, Remembering, Former died, sad hear passing, RIP one, sad passed, obituary, hear passed, sorry hear, legacy, hear great, condolences family, singer, Away, Dies via, battle, Passes, Sad passing, memory, mourns, RIP great, inspiration, sad death, news passing, deeply saddened, demise, Rest peace, soul, kind, news passed, Former away, Passes Away
Creative work	cancelled, canceled, Conference, Press Conference, cancels, ending, Cancels, Canceled, Cancelled, Parody, series finale, added video playlist, Finale, Heirs Parody, sad, hour, Master Sun Parody, Master Sun Heirs, Queen, Saw, PIC, Hit, seasons, press, earlier, last episode, Master Sun Heirs Parody, Arsenio, Chris Bosh Judge Joe, Chris Brown, Sword, Hit Us, Dianna, Dianna Hit Us, NEWS, Aqua Teen, Wet Hot American, new season, conference, Turn, Queen watched Lizzie Bennet, Underground, CNN, Patti, Broke Girls, released, Patti Smith, PIC Press, Like Follow, Like Retweet
Location	closing, end, demolition, Technical, Elementary, Sandy, Hart, Technical Institute, closes, years, building, close end, closed, Demolition, demolished, old, Negros, City, implosion, shut, doors, College close, roof, final, South, Candlestick Park, good, school, College closing, Courson, campuses, iconic, day, home, Music, closure, Law, bus, Kroger, Hopkins, live, Murder Kroger, Newseum building
Group	RIP, shutting, bankruptcy, operations, split, Sad, shuts, officially, breaking, broke, shut, break, closes, filed, buy billion, died, broken, administration, closed, Pfizer, dies, Billion, franchise, merge, gone, Oracle, closing stores, following, files bankruptcy, buy deal, crash, leaving, statement, suspends, End, startup, developer, duo, billion deal, Bankruptcy, Final, cease, closure, chain, cancer, close stores, dead, quits, confirms, calling
Event	Canadian Women, Canadian Hockey, Canadian Hockey League, Abu, BCS Championship, operations, cancelled, wins Grand, Final, LIVE, Crawl, Vettel Grand, winning, Hall Game Awards, Sebastian, Congratulations, Sebastian Vettel, Squamish, Squamish Music, Hockey Champions, Squamish Valley, second, Mark, set, Squamish Music Festival, Palestine, Awards, State, BCS Championship Game, Congrats, Sebastian Grand, Johnson, due, added, Sebastian Vettel Grand, India, FedEx, start, WWE 2017, Vettel Grand Prix, Pizza Bowl, got, good, victory, Football operations, Watch
Service &Product	shutting, closing, RIP, sad, print, shuts, Sad, hear, Shutting, Shuts, final, officially, Shut, publication, website, Goodbye, End, March, Gothamist, digital, announced, Live Messenger, ending, edition, cease, Rogers, closes, ends, stop, dead, June, death, run, August, closure, EA, Mini, September, goodbye, Microsoft Live, Concert, Service, longer, killing, Microsoft Windows Live, staff, million, YouTube, plug, announces

Table 4.5: Extracted patterns of the English disappearing contexts in the training data.

TYPE	extracted patterns
PERSON	死去, 冥福, 訃報, お祈り, 冥福 お祈り, 氏 死去, 死去 ニュース, 合掌, 死去 し, 冥福 し, お祈り し, 冥福 お祈り し, Yahoo, Yahoo ニュース, 亡くなり, 訃報 死去, 逝去, 冥福 いたし, 死去 朝日新聞, お祈り いたし, 冥福 お祈り いたし, 死去 Yahoo, 死去 Yahoo ニュース, 亡くなっ の, し 死去, 朝日新聞 デジタル, 死亡, 死去 デジタ ル, 死去 朝日新聞 デジタル, 亡くなら, し 冥福, 亡くなら れ, 作家 死去, 申し上げ, 訃報 し, れ 冥福, し お祈り, し 冥福 お祈り, 死去 死去, れ お祈り, れ 冥福 お祈り, がん, 冥福 死去, 亡くなっ し, 安らか, 氏 ニュース, お祈り 死去, 冥福 お祈り 死去, R I P, い 冥福
LOCATION	閉館, 歴史, 幕, 最後, 歴史 幕, 閉館 し, 思い出, 閉鎖, 駅 廃止, 別れ, 月末, 館 閉館, Yahoo, 解体, 劇場 閉館, 閉館 の, 閉店 し, ファン, 聖地, Yahoo ニュース, 閉館 する, 年間, 廃止 駅, ラスト, 駅 最後, 閉館 幕, 廃, RT 最終, 時代, 最終 列車, 最終 駅, 幕 ニュース, 閉館 閉館, 閉館 歴史, 閉館 館, 今日 の, 駅 最終, 歴史 ニュース, 最後 し, 閉館 歴史 幕, 歴史 幕 ニュース, 座 閉館, 今日 閉館, 閉店 の, 建物, 閉館 映画, お世話, 惜しむ, 今日 最後, 今日 駅
GROUP	解散, 合併, 解散 発表, 解散 し, ラスト, 休止, 活動 休止, 破産, 解散 する, 吸収, 解散 解散, ラスト ライブ, 解散 ニュース, 解散 ライブ, 解散 の, Yahoo ニュース, 解散 活動, 吸収 合併, し 解散, 脱退, 解散 ラスト, 合併 し, 破産 開始, 活動 終了, 活動 発表, 解散 ん, 芸能, 経営, Y ニュース, 消滅, 期限, ソフトバンク モバイル, ライブ 解散, し ニュース, 期限 活動, し 会社, ソフトバンク 合併, 解散 し し, 期 限 休止, 期限 活動 休止, モバイル 合併, ソフトバンク ワイ モバイル, 開始 決定, 引退, バンド 解散, 年間, ソフトバンク モバイル 合併, モバイル モバイル, 解散 い, ワイ モバイル 合併
EVENT	終了, 最後, フィナーレ, ブース, アニメ 2013, 最高, 新聞 大会, 華, 国際 女子 マラソン, 一周, 横浜 マラソン, 神奈川 新聞, 休止, 最終, ハーフマラソン, 東京 湾 華, 大会 終了, 花火 大会 休止, メンバー, 神奈川 新聞 花火 大会 休止, 東京 2013, 会場, 男子, U 日本, U 日本 代表, ブック フェア, 大濠 花火, 大濠 花火 大会, 大会 U, アニメ コンテンツ エキスポ 2013, 駅伝 終了, 本日 し, 決勝 大会, 東京 ブース, 競技 日本, 九州 一周 駅伝 終了, 九州 終了, 財政難, U 20 代表, 夢 大陸, 視聴, TOKAI, SUMMIT, 競技 大会 U, U 20 日本, U 20 日本 代表, 東アジア 大会 U, アニメ ステージ, 大会 U 代表, 競技 大会 代表
SERVICE &PRODUCT	サービス 終了, 号 休刊, 雑誌 休刊, 終了 お知らせ, 休刊 し, 休刊 雑誌, 幕, コミック 休刊, 終了 し, 破産, 歴史 幕, 休刊 休刊, サービス お知らせ, サービス 終了 お 知らせ, 休刊 の, 休刊 連載, 終了 ニュース, 廃刊, オンライン 終了, 残念, 発売 休 刊, まんが 休刊, 休刊 作品, 破産 ニュース, 歴史 幕 ニュース, サイト 閉鎖, 消滅, 休刊 発表, 休刊 コミック, 最終 号, 最後, 休刊 する, 発売 号 休刊, 閉鎖, 悲報, 休刊 歴史, Yahoo ニュース, サービス し, サービス 終了 し, 移籍, 終了 の, 終わる, Yahoo 終了, 休刊 幕, 休刊 歴史 幕, 休刊 ん, 休刊 ニュース, 終了 サービス, サービ ス サービス, 次号 休刊

Table 4.6: Extracted patterns of the Japanese disappearing contexts in the training data.

4.5.2 Models

The following models were implemented for comparison:

Proposed (TDS + RefEmb): We implemented LSTM-CRF with flair embeddings [6] and proposed refined embeddings (RefEmb) using the training data constructed by proposed time-sensitive distant supervision (TDS). We refined pretrained fastText embeddings for each input post using tweets on the day of the input post (about 1 to 2M tweets for each day) and additionally fed them into the LSTM-CRF.

Proposed (TDS): To verify the effectiveness of refined word embeddings, we implemented LSTM-CRF with only flair embeddings [6] using the same training data, optimization and parameters as Proposed (TDS + RefEmb). This method does not consider the multiple posts of the target day when recognizing entities.

Baseline: To verify the effectiveness of the constructed training data, we collected posts using the original version of time-sensitive distant supervision and trained models with that data. Specifically, we changed only the timestamps considered in the original method in Step 2 of § 4.4.1 for each entity and collected 100 reposts (retweets) of the last day in which the entity appeared more than 10 times as positive examples. As for negative examples, we obtained the same number of posts from more than one year before the date when we collected the positive examples. By using the collected 25,920 posts for 2,867 entities in English and 6,777 posts for 1,733 entities in Japanese, we trained LSTM-CRF with flair embeddings using the same optimization and parameters as Proposed (TDS + RedEmb) and applied it to the test data. Since this method does not consider the year of the disappearance of entities, it may collect many noisy contexts.

4.5.3 Settings

We tokenized each input post using spaCy (ver. 2.0.12)[7] with en_core_web_sm model for English and using MeCab (ver. 0.996)[8] with ipadic (ver. 2.7.0) for Japanese.

[7]https://spacy.io
[8]https://taku910.github.io/mecab

Parameter	Value
Character embedding size (LM)	30
Dimension of Character Bi-LSTM (LM)	1024
SGD learning rate (LM)	20.0
Batch size (LM)	100
Word embedding size (LSTM-CRF)	300
Dimension of Word Bi-LSTM (LSTM-CRF)	256
Batch size (LSTM-CRF)	32
Dropout (LSTM-CRF)	0.5
SGD learning rate (LSTM-CRF)	0.01

Table 4.7: Hyperparameters of character-based language model (LM) and LSTM-CRF.

We use Keras (ver. 2.3.1)[9] for implementing all the models. For flair embeddings, we set hyperparameters as suggested in [6] and trained the character-based bidirectional LSTM language model from 2B English tweets for English and 800M Japanese tweets, respectively, both posted from March 11th, 2011 to December 31st, 2011. We show the hyperparameters in Table 4.7. Using the same tweets, we trained 300-dimensional word embeddings using fastText[10] and used them for initializing the embedding layers by concatenating with flair embeddings. We optimized all the models using stochastic gradient descent and chose the model at the epoch with the highest F1-score on the development data.

4.5.4 Results and Analysis

Overall performances of the models To evaluate the discovery of disappearing entities, we apply the models to each post in the test data constructed in § 4.5.1 and evaluate the results using the CoNLL-2003 [67] schema, which measures the performance of the models in terms of precision, recall, and F1-score. Table 4.8 shows the micro and macro precision, recall, and F1-score for all the models. We see that our proposed methods both outperformed the baseline, which collected training data without considering the year of disappearance for the entities. The very low performance of the baseline is because it trained with the training

[9]https://keras.io
[10]https://fasttext.cc/

	Precision		Recall		F1	
	micro	macro	micro	macro	micro	macro
Proposed (TDS + RefEmb)	0.730	0.668	**0.671**	**0.545**	**0.699**	**0.592**
Proposed (TDS)	**0.766**	**0.691**	0.587	0.437	0.665	0.522
Baseline (TDS)	0.514	0.491	0.184	0.219	0.271	0.284

(a) English

	Precision		Recall		F1	
	micro	macro	micro	macro	micro	macro
Proposed (TDS + RefEmb)	**0.850**	**0.671**	**0.599**	**0.479**	**0.708**	**0.567**
Proposed (TDS)	0.828	0.650	0.532	0.425	0.648	0.513
Baseline (TDS)	0.743	0.556	0.143	0.123	0.240	0.196

(b) Japanese

Table 4.8: Overall performances of each method for English and Japanese.

data that contains much noise. This shows that our improved time-sensitive distant supervision properly collected the contexts of disappearing entities. Our Proposed (TDS + RefEmb) achieved the best performance, which means that the proposed refined word embeddings worked effectively. In particular, the recall was improved in both Japanese and English, which indicates that entities that could not be recognized by only using the features of a single post can be successfully detected by utilizing multiple posts.

Detailed performances of the best performing model To analyze the behavior of our proposed model, we show the precision, recall, and F1-score of Proposed (TDS + RefEmb) for each coarse-type in Table 4.9. We see that the performance of PERSON type entities is high in both Japanese and English. This is probably because a large number of entities of this type in the training data and the person's name itself is easy to recognize from the entity's surface. The performance of CREATIVE WORK is the lowest in English. This is probably because the disappearance of this type of the entities is uncertain for the nature of the type. For example, even if the final episode of a television drama is aired on TV, it remains in various media. This causes the training data to be contaminated with diverse contexts, resulting in the poor performance of the model. We also see that the performance of EVENT type entities is the lowest in Japanese. This may be because the number of data in the training data is small, and thus the model could not be sufficiently trained.

	Precision	Recall	F1	#NE
Person	0.865	0.901	0.883	426
Creative work	0.480	0.500	0.490	24
Location	0.727	0.491	0.586	114
Group	0.570	0.526	0.547	272
Service&Product	0.566	0.409	0.475	115
Event	0.800	0.444	0.571	18

(a) English

	Precision	Recall	F1	#NE
Person	0.948	0.858	0.901	233
Location	0.814	0.569	0.670	123
Group	0.818	0.418	0.553	194
Service&Product	0.777	0.552	0.645	145
Event	0.250	0.042	0.071	24

(b) Japanese

Table 4.9: Detailed perfomances of Proposed (TDS + RefEmb)

Relative recall and detection immediacy To evaluate the detection immediacy of our method, we refer to the experiment of relative recall in § 3[4]. Specifically, for Wikipedia entities that disappeared in 2019 (2,608 for English and 763 for Japanese), we applied Proposed (TDS + RefEmb) to all the posts in 2019 in which each entity appeared (437,816 for English and 202,666 for Japanese) and checked how many entities we could discover from them. For the detection immediacy, we check how early we can discover the disappearing entities from the timing of the update when the disappearance is added to the category in each entry in Wikipedia.

Table 4.10 and 4.11 show the distribution of the types of target entities for both languages obtained by the Wikipedia categories, detection ratio, and lead-time against the Wikipedia update time. Overall, Proposed (TDS + RefEmb) detected 2,017 (77.34%) English and 582 (76.28%) Japanese disappearing entities. More than 80% of entities of Person are detected in both languages, while the other types are found only about 30% to 60%. Note that some target entities are low frequency on our Twitter archive and do not appear in disappearing contexts. Since Proposed (TDS + RefEmb) utilizes such disappearance signals as the clue, it is difficult to

TYPE	# entities	# found (%)	lead-days mean (median)	
Person	1838	1668 (90.75%)	21	(0)
Creative work	351	123 (35.04%)	173	(64)
Location	73	38 (52.05%)	150	(46)
Group	163	99 (60.74%)	195	(131)
Service&Product	93	47 (50.54%)	172	(91)
Event	25	8 (32.00%)	85	(35)
Unmapped	65	34 (52.31%)	237	(171)
Total	2608	2017 (77.34%)	49	(0)

Table 4.10: Relative recall and time advantage over entity types of English disappearing entities detected with Proposed (TDS + RefEmb).

TYPE	# entities	# found (%)	lead-days mean (median)	
Person	515	446 (86.60%)	31	(0)
Location	48	31 (64.58%)	149	(117)
Group	121	56 (46.28%)	181	(147)
Service&Product	63	43 (68.25%)	154	(98)
Event	16	6 (37.50%)	183	(171)
Total	763	582 (76.28%)	63	(0)

Table 4.11: Relative recall and time advantage over entity types of Japanese disappearing entities detected with Proposed (TDS + RefEmb).

Entity: **Barry Hughart** True type: PERSON
Oh well crap! RIP **Barry Hughart**. Bridge of Birds is still a favorite.

Entity: **BetBright** True type: GROUP
I genuinely hope all the staff will find new jobs. In memory of **BetBright**. Poem idea care of USER

Entity: **WikiTribune** True type: SERVICE&PRODUCT
'Jimmy Wales' **WikiTribune**, which launched as a crowdfunded news site in 2017, relaunches as WT : Social , a donor - fund

Entity: **ArenaBowl** True type: NULL
SOUL WELCOME BACK WR LARRY BRACKINS: The Philadelphia Soul announce the re-signing of **ArenaBowl** XXII Champion wi... URL

Table 4.12: Examples that our model predicted correctly (above two) and incorrectly (below two) (English)

discover entities appearing without disappearing contexts. Considering this, our method can cover the disappearing entities at a reasonable rate.

About the detection immediacy, we found that 77.3% of the discovered English entities (1,560 out of 2,017) and 85.4% of the discovered Japanese entities (497 out of 582) were detected earlier than their update in Wikipedia. The mean (and median) lead days of the first day when Proposed (TDS + RefEmb) detected each entity against their update date were 49 (and 0.13) for English and 63 (and 0.44) days for Japanese, which supports the detection immediacy of our method. It is interesting to note that the update of PERSON type entities is surprisingly faster for both languages. This indicates that only certain types of updates about entities are made quickly in Wikipedia. For the remaining types, we detected them more than a month or several months earlier, indicating the promptness of our method.

Examples Finally, we show the examples of predictions with Proposed (TDS + RefEmb) in Table 4.12. Even when the length of the post is short, and there are few clues as in the first example, or when the disappearance is not directly mentioned as in the second example, Proposed (TDS + RefEmb) recognized entities correctly by using the features obtained from other multiple posts in the Twitter stream. On the other hand, like the third example, Proposed (TDS + RefEmb) failed to detect the disappearing context containing rare words such as 'relaunch,' which do not appear in the training data. We also found that some false positive predictions were caused by words that are often found in disappearing contexts, such as 'announce' in the fourth example.

4.6 Chapter Summary

We introduced a novel task of discovering disappearing entities in microblogs (§ 4.1, 4.2 and 4.3). To deal with disappearing entities, where their disappearance is difficult to recognize, we explicitly considered the year of disappearance in Wikipedia and fed it into the time-sensitive distant supervision method (§ 4.4.1). We proposed the method of refining pretrained word embeddings, which is utilized to robustly recognize disappearing entities, using the Twitter stream of the day of the input post (§ 4.4.2). Experimental results demonstrated that our discovering method with the constructed dataset outperformed the baseline method and successfully found more than half of the target disappearing entities in Wikipedia earlier than the update of the disappearance (§ 4.5).

Chapter 5

Typing of Emerging Entity

5.1 Introduction

Microblogs enable us to instantly share a wider variety of topics than news streams [30] and have become one of the primary sources for acquiring new information. To analyze this massive volume of posts for applications such as social-trend analysis and entity recommendation, it is necessary to extract entity units from them and classify their types using techniques such as named entity recognition (NER) and entity linking [73]. However, newly 'emerging' entities (*e.g.*, Avatar 2) are difficult to handle because they do not exist in the training data of supervised models or the knowledge bases (KBs), and valuable information of the entities is often thrown away.

Motivated by this background, we defined emerging entities as those which appear in contexts that emphasize their novelty and attempted to discover emerging entities from microblogs (§ 3)[4]. To extract emerging entities, we exploited the fact that entities appear in characteristic contexts when they first emerge (*e.g.*, new games often appear with "trailer," "release" and a console name (Figure 5.1)), and developed a method of discovering them from microblogs. Although our method detected emerging entities promptly, typing those emerging entities is still necessary for downstream applications.

Existing studies on entity typing, however, focus on non-emerging (or prevalent) entities [44, 70, 76, 60, 7]. Most of them classify single mentions of entities into their context-dependent types. To complement a scarce context, many studies

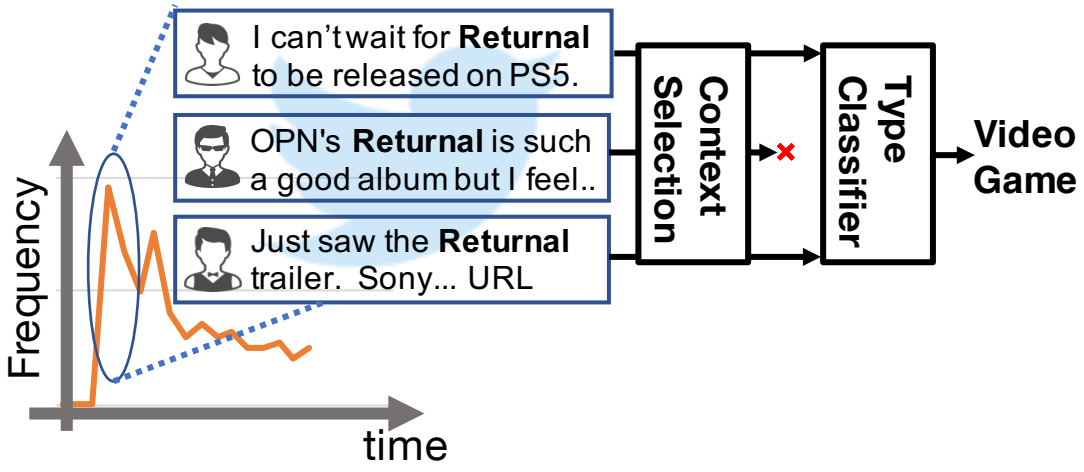

Figure 5.1: Emerging entity typing: identify the type of a given emerging entity with its first burst of posts.

rely on language resources such as KBs to narrow down the candidate types. Unfortunately, those resources are not available for newly appearing entities. It is thus unrealistic to perform accurate mention-level typing using these methods in a short and noisy microblog post.

We thus design a task of identifying a fine-grained entity type from a burst of posts about the target entity (Figure 5.1), assuming that the target mention is detected in advance. This batch-level typing is a more realistic setting for emerging entities than the conventional mention-level typing.

To build training data for this task, we collect emerging entities and their contexts for English and Japanese using time-sensitive distant supervision (§ 3)[4]. To evaluate typing methods, we manually build test data for two types of emerging entities: homographic and non-homographic; homographic entities share names with other words (*e.g.*, 'Go' for a board game, a programming language, and a verb) and consequently their contexts are contaminated.

We then propose a modular entity typing model that performs multi-instance (MI) learning [64, 78]. In addition to contexts for the entity and its entity surface, this model leverages meta-information such as URLs and usernames by exploiting the characteristics of the microblog domain. Because entities can have homographs, it is risky to use all the posts obtained using simple string matching as contexts for typing. We thus propose to find and use emerging contexts since

two emerging entities with the same name are unlikely to emerge in a short period of time, and such contexts are useful for typing.

We finally evaluate our typing model on the above English and Japanese Twitter datasets. Experimental results confirm that our model outperforms a baseline model that performs MI-learning with randomly selected posts in training and testing. We demonstrate that when typing homographic emerging entities, it is more important to selectively use emerging contexts and meta information.

Our contributions are as follows:

- We set up a task of fine-grained typing of emerging entities in microblogs.
- We built two large-scale Twitter datasets for English and Japanese. We will release them to facilitate future studies.
- We proposed an entity typing model and a context selection model that outperformed a baseline with MI-learning.

5.2 Related Work

In this section, we first review existing studies on the definition and detection of emerging entities. We then explain the existing task settings of entity typing and discuss their limitations.

5.2.1 Emerging Entity Detection

Although there are studies that find "emerging" entities [54, 34, 75, 22], most of them in fact consider out-of-KB entities, which include not only emerging entities that are not prevalent (newly appeared and yet not widely known) in the world but also prevalent entities that are absent from the incomplete KBs such as Wikipedia. Although we do not handle prevalent out-of-KB entities in this study, we intend to type those entities before they become prevalent in a microblog.

To target only truly emerging entities, we defined emerging entities as those which appear in emerging contexts that emphasize their novelty (§ 3)[4]. With this definition, they developed a time-sensitive distant supervision method,

which uses time-stamps of microblogs to collect early posts (contexts) in which non-homographic KB entries (entities) appear. Using the datasets collected for Japanese, they trained an emerging entity recognizer, which successfully discovered various emerging entities more than one year before their registrations into Wikipedia.

In this study, we adopt the definition of emerging entities (§ 3)[4] and conduct time-sensitive distant supervision to automatically construct large-scale English and Japanese Twitter datasets for typing emerging entities.

5.2.2 Entity Typing

Traditionally, named entity recognition [67, 65, 74, 47, 5, 71, 72] jointly performs recognition and typing of entity mentions in the text. However, most NER models require costly training data that fully annotate all entities in the text. Indeed, many studies adopt less than ten coarse types (*e.g.*, person, location, and organization) [48].

Focusing on fine-grained entity typing, recent studies adopted distant supervision [52] that automatically annotates entities with KB categories, and tackled the task of classifying single mentions of entities with their types in a context [44]. This allows us to exploit resource-hungry neural models [70] and knowledge of the target entity derived from KBs [60, 76] or a large corpus [20]. Although these methods succeeded in mitigating context scarcity and typing entities accurately, they are not effective when typing emerging entities that are absent from the KBs and the corpus.

To enumerate all possible types for out-of-KB entities, Lin et al. [42] and Nakashole et al. [54] performed entity-level entity typing (as multi-label classification). They extracted local contexts (patterns) from multiple sentences (contexts) in which entities appeared and propagated types from in-KB entities that exhibit similar patterns. However, this approach needs massive contexts to obtain reliable patterns. Yaghoobzadeh et al. [78] and Xu et al. [77] elaborate on these methods by using embeddings of entities instead of patterns and by encoding actual contexts with a neural network. However, this approach cannot be directly used to type emerging entities since it is difficult to collect contexts for emerging entities; the

entity linking they used to collect contexts requires KBs that are not available for emerging entities.

In this study, to type emerging entities in a microblog as early as possible, we set up a task of entity-level fine-grained typing of emerging entities from a burst of posts (§ 5.3.1). We build Twitter datasets for this task (§ 5.3.2) and develop an effective typing method (§ 5.4).

5.3 Task and Datasets

This section defines our task of typing emerging entities, and then we describe our dataset for this task.

5.3.1 Task Settings

Inspired by the related studies on entity typing (§ 5.2.2) and the definition of emerging entities, we design the task of emerging entity typing. We take the following points into consideration: 1) For applications such as social trend analysis, we want to type emerging entities as soon as they appear. 2) Since microblog posts are short and noisy, we practically need more than one post for accurate typing. In fact, the accuracy of Twitter NER is very low (29.7%) for out-of-vocabulary entities [29]. 3) Emerging entities show an early burst of posts around the time of their introduction into public discourse [30]. These considerations lead us to the following task settings: Given an entity and a burst of posts containing the entity, the goal of the task is to predict the single type of entity as multi-class classification. We assume a single type for emerging entities since two entities with the same name are unlikely to simultaneously emerge in a short period of time. As for the burst, to simplify the task, we split posts by a day defined by the UTC-0 time zone and considered a burst to have occurred if an entity string appeared more than 10 times in any of the bins for the first time.

There are two challenges in this task: 1) How to perform accurate typing in situations where we cannot assume the existence of emerging entities in language resources such as KBs and massive contexts. 2) How to deal with homographic emerging entities where a simple string match would cause contamination of contexts for the target entity.

5.3.2 Dataset Construction

We construct training, development, and test data for our task, following the above definition and the task settings. We adopt Twitter as a microblog and target English and Japanese, which are the top two languages on Twitter [8]. We use our archive of Twitter posts that are retrieved[1] by using the official Twitter APIs[2] and consists of more than 50B posts (32% are English and 20% are Japanese; This does not deviate much from the actual data [8]). In the following, we explain how we automatically create training and development data and how to manually build the test data for non-homographic and homographic emerging entities.

Training Data

To create the training data, we used time-sensitive distant supervision (§ 3)[4] to collect the contexts of entities in Wikipedia at the time they emerge. For both English and Japanese, we gathered the titles of articles as candidates of emerging entities registered in Wikipedia from Mar. 11th, 2012 to Dec. 31st, 2015. To remove entities that may not be emerging, we discarded the titles that were not reposted more than 10 times or more. Since the entity string (*e.g.*, 'Go') may refer to multiple entities (a programming language and a board game) and existing words (verb), we discarded the titles that appeared 10 times in the period of Mar. 11th, 2011 to Mar. 10th, 2012 to avoid contamination with non-emerging contexts.[3]

Next, we retrieved all posts for the period from Mar. 11th, 2012 to Dec. 31th, 2019 where each of the collected entities appeared in our Twitter archive. Using these data, we collected 50 posts up to the date of the first burst of each entity as emerging contexts. We collected another 50 posts for each entity one year after the time of the initial collection as prevalent contexts. We used these contexts as negative examples of a context selection model and pretraining the typing model.

[1]Starting from 26 popular Japanese users in Mar. 2011, their timelines (recent tweets) have been continuously collected using user_timeline API, while the user set has iteratively expanded to those who were mentioned or whose tweets were reposted by already targeted users.

[2]https://developer.twitter.com/en/docs/twitter-api

[3]If the entities (*e.g.*, programming language, Swift) appear long before (here, from 2011 to 2012) their registrations into Wikipedia (here, June 2nd, 2014), their names may not be unique and can have non-emerging homographic entities (e.g., person, Taylor Swift).

We mapped the collected entities to their corresponding fine-grained types assigned in the DBpedia [9] ontology; for example, the entity "Spider-Man: Homecoming" is mapped to the type "Film." For analysis purposes, we manually classified the mapped types into coarse-grained types for each language derived from § 3[4]. As a result, we obtained 597,569 emerging contexts and 859,034 prevalent contexts from 37,374,820 posts for 20,571 entities with 6 coarse-grained and 185 fine-grained types for English. For Japanese, we obtained 259,484 emerging contexts and 435,499 prevalent contexts from 47,869,813 posts for 10,315 entities with 4 coarse-grained and 71 fine-grained types. The difference in the number of types comes from the degree of DBpedia development for each language.

Table 5.1 shows the statistics of obtained emerging entities and contexts. We see that the frequency of fine-grained types varies by language; for example, the English Person type includes many athletes entities, while the Japanese Person type does not. This reflects the fact that the coverage of entities in Wikipedia varies across languages.

Test Data

For non-homographic emerging entities, we built the test data similarly to the training data and then manually cleaned the data for reliable evaluation. Specifically, we collected the titles of Wikipedia articles as entities that appeared more than 100 times on our Twitter archive from Jan. 1st, 2017 to June 20th, 2018 for English and from Jan. 1st, 2016 to June 20th, 2018 for Japanese. We then collected posts up to the date of the first burst for each entity. Since those entities may not actually be emerging, we removed entities whose posts are judged to include only prevalent contexts by two of three annotators (the first author and two graduate students). We obtained an inter-rater agreement of 0.782 for English and 0.771 for Japanese by Fleiss' Kappa [27]; both show substantial agreement. We finally obtained 31,244 posts for 1200 emerging entities in English and 16,869 posts for 800 emerging entities in Japanese, each containing 200 entities of each coarse-grained type. Table 5.3 show the statistics of the data.

For homographic emerging entities, we manually constructed the test data since it is difficult to collect their contexts using distant supervision. We collected

TYPE DBpedia types	#ent.	#posts
PERSON	**9878**	**316123**
Person	2514	73517
SoccerPlayer	1337	41955
AmericanFootballPlayer	1157	43737
Politician	1017	27626
BaseballPlayer	560	22439
Others (68 types)	3293	106849
CREATIVEWORK	**6979**	**192214**
Film	1777	46185
Album	1272	31947
TelevisionShow	1043	26526
VideoGame	946	30895
Single	698	21272
Others (20 types)	1243	35389
LOCATION	**1588**	**31554**
City	912	14566
Building	146	3922
Stadium	66	2260
Settlement	51	1636
Convention	43	1313
Others (31 types)	370	7857
GROUP	**1413**	**39260**
Company	719	20148
Organisation	223	6172
Band	137	3745
SoccerClub	81	2440
PoliticalParty	62	1711
Others (18 types)	191	5044
DEVICE	**335**	**9404**
Device	147	4053
Automobile	69	2100
InformationAppliance	66	2058
Ship	31	657
Weapon	14	306
Others (3 types)	8	230
EVENT	**378**	**9014**
Award	110	2593
SpaceMission	46	910
MixedMartialArtsEvent	46	1253
MilitaryConflict	26	540
HorseRace	23	727
Others (15 types)	127	2991
TOTAL	**20571**	**597569**

(a) English data

TYPE DBpedia types	#ent.	#posts
PERSON	**3995**	**105207**
Actor	729	18506
MusicalArtist	567	16149
SoccerPlayer	419	10621
VoiceActor	383	7169
BaseballPlayer	365	10978
AdultActor	242	8021
Politician	237	5840
Model	230	7670
Person	146	3679
Writer	118	2742
Wrestler	118	2747
Others (17 types)	441	11085
CREATIVEWORK	**5706**	**140191**
Single	1211	28985
TelevisionShow	1058	26488
Album	842	18436
Film	799	20075
VideoGame	562	18587
Manga	496	10357
Anime	292	7224
Radio	250	4919
Book	122	3184
Song	28	594
Others (4 types)	46	1342
LOCATION	**304**	**6419**
Building	98	2421
Museum	33	775
Station	32	694
Settlement	23	376
School	20	224
City	17	355
Mountain	13	221
University	10	183
Park	8	118
Hospital	7	150
Temple	6	113
Others (14 types)	37	789
GROUP	**310**	**7667**
Company	216	5373
SoccerClub	48	1075
Organisation	24	642
PoliticalParty	22	577
TOTAL	**10315**	**259484**

(b) Japanese data

Table 5.1: Statistics of emerging entities and a burst of posts in the training data obtained from Twitter.

Entity: **Star Wars: The Force Awakens** Type: Film
1. **Star Wars: The Force Awakens** has completed principal photography. HASH HASH URL
2. Wow! 3 words! Yes! RT USER: The official title for Episode VII is '**Star Wars: The Force Awakens**.' URL
3. **Star Wars: The Force Awakens**. My cynical side has nothing for that, so I guess I'm happy with the title.

Entity: **Ben Sheaf** Type: SoccerPlayer
1. Arsenal have made England youth midfielder **Ben Sheaf** their first signing of the summer.
2. Arsenal sign **Ben Sheaf** from West Ham URL
3. Who is **Ben Sheaf**?

Entity: **Another Life** Type: TelevisionShow
1. RT USER: Here are a few titles in the upcoming HASH: In **Another Life** || Fall of the Planet of the Apes || Terms & Conditions || Are. . .
2. **Another Life** - Netflix Orders Space Drama Starring Katee Sackhoff (Posted: 2018-04-26 13:40:48). . .
3. RT USER: Now playing **Another Life** by lightcraft! Check it out: URL

Table 5.2: Examples of the **emerging entities** and a burst of posts. The third example is a homographic entity.

the titles of Wikipedia articles, each of which has a disambiguation page, and gathered the newest one with their posts from the same period. Since those entities share contexts with other entities of the same name, we asked the three annotators to identify the exact day when the target entity first appears with emerging contexts for the given type. We adopt entities with the answers (days) agreed upon by two or more annotators. We obtained an inter-rater agreement of 0.684 for English and 0.665 for Japanese by Fleiss' Kappa; both show substantial agreement. We collected the posts of that day and the previous day and finally got a total of 5,931 posts for 200 emerging entities in English and 13,430 posts

TYPE DBpedia types	#ent.	#posts
PERSON	**200**	**6048**
SoccerPlayer	51	1447
Politician	36	875
Person	29	839
AmericanFootballPlayer	15	518
IceHockeyPlayer	11	475
Others (22 types)	58	1894
CEATIVEWORK	**200**	**5327**
Album	50	1280
Film	40	1097
TelevisionShow	37	750
VideoGame	28	948
Book	12	299
Others (11 types)	33	953
LOCATION	**200**	**5687**
Stadium	38	980
Building	35	1337
Museum	19	424
Station	15	554
School	13	522
Others (24 types)	80	1870
GROUP	**200**	**4907**
Organisation	44	1077
PoliticalParty	36	808
Company	33	942
SoccerClub	18	407
MilitaryUnit	14	184
Others (11 types)	55	1489
DEVICE	**200**	**5302**
Device	76	2070
Automobile	45	1343
Ship	35	834
InformationAppliance	18	537
Weapon	17	315
Others (3 types)	9	203
EVENT	**200**	**3973**
Award	61	1064
GrandPrix	21	443
WrestlingEvent	14	340
MixedMartialArtsEvent	12	261
HorseRace	12	295
Others (13 types)	80	1570
TOTAL	**1200**	**31244**

(a) English data

TYPE DBpedia types	#ent.	#posts
PERSON	**200**	**4149**
SoccerPlayer	40	765
Politician	22	310
Presenter	21	455
Actor	19	444
AdultActor	17	335
BaseballPlayer	13	210
Wrestler	12	195
MusicalArtist	11	387
VoiceActor	10	198
Model	8	266
Writer	7	121
Others (6 types)	20	463
CREATIVEWORK	**200**	**4058**
Manga	36	391
TelevisionShow	33	902
VideoGame	32	948
Film	26	410
Single	25	436
Album	21	360
Anime	15	366
Radio	8	213
Book	3	26
Song	1	6
LOCATION	**200**	**4203**
Building	79	1978
Station	51	934
Museum	26	437
Library	9	159
School	8	169
Infrastructure	6	47
University	5	60
ArchitecturalStructure	5	149
Park	4	104
RailwayLine	3	46
Hospital	2	49
Others (2 types)	2	71
GROUP	**200**	**4459**
Company	168	3859
PoliticalParty	14	240
SoccerClub	12	275
Organisation	6	85
TOTAL	**800**	**16869**

(b) Japanese data

Table 5.3: Statistics of **non-homographic** emerging entities and a burst of posts in the test data obtained from Twitter.

TYPE DBpedia types	#ent.	#posts
PERSON	**65**	**1750**
AmericanFootballPlayer	13	448
SoccerPlayer	10	255
MartialArtist	9	212
BasketballPlayer	9	221
Politician	8	195
Person	6	243
Wrestler	3	42
RugbyPlayer	3	95
Boxer	2	16
Model	1	12
IceHockeyPlayer	1	11
CREATIVEWORK	**125**	**3892**
TelevisionShow	27	861
Film	22	547
Single	19	800
VideoGame	15	560
Album	14	453
Book	12	256
Comic	5	183
Song	4	94
Anime	3	51
Website	1	15
Software	1	50
Musical	1	9
Manga	1	13
LOCATION	**2**	**100**
Stadium	1	50
Building	1	50
GROUP	**6**	**89**
PoliticalParty	3	14
Company	3	75
DEVICE	**1**	**50**
InformationAppliance	1	50
EVENT	**1**	**50**
WrestlingEvent	1	50
TOTAL	**200**	**5931**

(a) English data

TYPE DBpedia types	#ent.	#posts
PERSON	**38**	**1610**
MusicalArtist	11	735
ComedyGroup	7	336
AdultActor	4	53
Actor	3	153
VoiceActor	2	102
SoccerPlayer	2	63
Model	2	46
BaseballPlayer	2	37
Wrestler	1	29
Presenter	1	10
Politician	1	17
Person	1	11
AthleticsPlayer	1	18
CREATIVEWORK	**156**	**11310**
Single	39	3002
Album	33	2481
Film	28	1959
TelevisionShow	20	1655
Manga	15	817
VideoGame	7	463
Anime	5	339
Radio	4	238
Song	2	133
Book	2	123
Software	1	100
GROUP	**6**	**510**
Company	6	510
TOTAL	**200**	**13430**

(b) Japanese data

Table 5.4: Statistics of **homographic** emerging entities and a burst of posts in the test data obtained from Twitter.

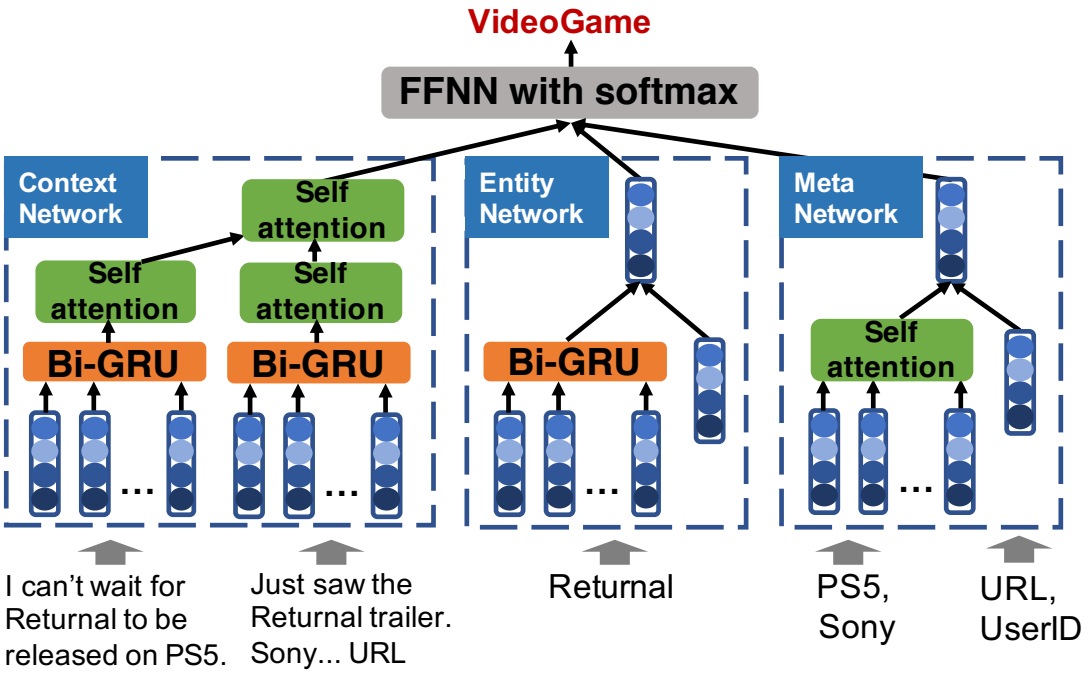

Figure 5.2: Overview of our entity typing model ($N = 2$): three networks process contexts, entity, and meta-information, respectively using MI-learning.

for 200 entities in Japanese (see Table 5.4 for the statistics). Table 5.4 show the statistics of the data. [4]

Table 5.8 shows some examples of collected entities and their posts (excerpts). The first example is a non-homographic emerging entity in the training data. The second example is a non-homographic emerging entity in the test data. From this example, we see that there is a useless context for guessing the type (*e.g.*, No. 3). The third example is a homographic emerging entity in the test data, and as we can see, it contains a noisy context (*e.g.*, No. 3) that is not related to the target entity. We thus have to properly select only the related contexts of the target entity to predict its type.

[4]The data is unbalanced because homographic names tended to be concentrated in certain types, such as names of people and creative works.

5.4 Proposed Method

This section presents a method for typing emerging entities in microblogs. Microblogs have the following characteristics: most posts are short and noisy, several posts about the same topic appear in close time series, and it has meta-information such as usernames and URLs that are useful for inferring the type. We thus develop a neural typing model based on diverse features and MI-learning (§ 5.4.1).

Considering the existence of homographic entities (*e.g.,* Go), one may want to select only the posts that are relevant to the target entity, rather than using all posts when performing MI-learning. We thus develop a context selection model that ranks emerging contexts of the target entity (§ 5.4.2). In the following, we describe the details of each model and how to train and test the models.

5.4.1 Entity Typing Model

To capture the characteristics of emerging entities from diverse perspectives, we develop a modular model that consists of three neural networks (Figure 5.2): Context Network and Entity Network that encode contexts and entities, which are based on Yaghoobzadeh et al. [78] while refining their classic CNN-based structure with GRU [11] and self-attention mechanism [43], and Meta Network that encodes meta-information specific to microblogs. We rely on MI-learning [64], which assigns a single label to a bag of multiple instances to increase the number of clues and to mitigate the effects of noise induced by distant supervision. The final prediction is made by feeding the output of each network into the softmax layer through a feed-forward network as:

$$\mathbf{p} = \text{softmax}(W_o[\mathbf{o}_{context}; \mathbf{o}_{entity}; \mathbf{o}_{meta}] + \mathbf{b}_o) \tag{5.1}$$

We describe the details of each network hereafter.

Context Network

This model captures contexts of given posts; it differs from the Context Model [78] in that we change CNN to GRU and introduce a self-attention mechanism to

capture longer relationships and dependencies between words [81]. Specifically, we encode the given entity using MI-learning by inputting N contexts where the entity appears. We convert each word $w_{it}, t \in [1, S_i]$ of the i-th input context to $\mathbf{x}_{it}$ using the embedding matrix W_w, $\mathbf{x}_{it} = W_w w_{it}$. We input this into a bidirectional GRU as $\mathbf{h}_{it} = \text{BiGRU}(\mathbf{x}_{it})$, and apply self-attention to the entire hidden states to capture the word relations:

$$\alpha_{ijk} = \frac{\exp(\sigma(W_u \mathbf{u}_{ijk} + \mathbf{b}_u))}{\sum_k \exp(\sigma(W_u \mathbf{u}_{ijk} + \mathbf{b}_u))} \tag{5.2}$$

$$\mathbf{u}_{ijk} = \tanh(W_h \mathbf{h}_{ij} + W_h \mathbf{h}_{ik} + \mathbf{b}_h) \tag{5.3}$$

$$\acute{\mathbf{h}}_{ij} = \sum_k \alpha_{ijk} \mathbf{h}_{ik} \tag{5.4}$$

We first obtain the similarity $\mathbf{u}_{ijk}$ between $\mathbf{h}_{ij}$ and $\mathbf{h}_{ik}$. We use additive attention that consists of a feed-forward network to calculate those alignment scores. We then compute the importance weight α_{ijk} using the softmax function. After that, we obtain $\acute{\mathbf{h}}_{ij}$ as a weighted sum of the hidden layers. These $\acute{\mathbf{h}}_{ij}$ are concatenated to form the sentence representation $\mathbf{s}_i = [\acute{\mathbf{h}}_{i1}; ...; \acute{\mathbf{h}}_{iS}]$.

Once we have N sentence representations, we apply self-attention to them again to get the relations between sentences:

$$\alpha_{ij} = \frac{\exp(\sigma(W_u \mathbf{u}_{ij} + \mathbf{b}_u))}{\sum_j \exp(\sigma(W_u \mathbf{u}_{ij} + \mathbf{b}_u))} \tag{5.5}$$

$$\mathbf{u}_{ij} = \tanh(W_s \mathbf{s}_i + W_s \mathbf{s}_j + \mathbf{b}_s) \tag{5.6}$$

$$\acute{\mathbf{s}}_i = \sum_j \alpha_{ij} \mathbf{s}_j \tag{5.7}$$

These $\acute{\mathbf{s}}_i$ are concatenated and used as output $\mathbf{o}_{context} = [\acute{\mathbf{s}}_i; ...; \acute{\mathbf{s}}_N]$.

Entity Network

This model captures a given entity surface; it differs from the Global Model [78], in that we change CNN to GRU and remove the KB embeddings of the target entity because they are not available for emerging entities. This model predicts

the type of the target entity from its sequence of characters and words. We convert each character $c_t, t \in [1, C_i]$ of the target entity to $\mathbf{x}_t$ using the embedding matrix W_c, $\mathbf{x}_t = W_c c_t$. Similarly to the Context Network, we input this into a bidirectional GRU and obtain the character-based entity representation as $\mathbf{h} = \text{BiGRU}(\mathbf{x}_t)$.

Tokens inside the entity name are also useful clues. We obtain a token-based entity representation $\mathbf{y}$ by simply taking the average of the pre-trained word embeddings $\mathbf{v}_j$ divided by the number of tokens V in the entity as $\mathbf{y} = \frac{\sum_j \mathbf{v}_j}{V}$. These representations are concatenated and used as output $\mathbf{o}_{entity} = [\mathbf{h}; \mathbf{y}]$.

Meta Network

In addition to the contexts and the entity name, meta-information such as URLs and user (author) information are useful for typing emerging entities in microblogs. For example, URLs (*e.g., https://blog.playstation.com/2020/12/10/returnal-launches-on-ps5-march-19-2021/*) often include clues of the entity type, and users like official accounts often post about a specific type of an entity (*e.g., @NintendoAmerica* often announces about their new game products). Moreover, we can extract, from KBs, useful knowledge on in-KB entities that co-occur with the target entity.

We thus extract the above meta information from the input N posts and convert them into a feature vector. For user information, we simply extract the author's user IDs. As for URLs, we extract all URLs from the input. For each URL, we discard the URL parameters after the '?' or '&', and then separate the remaining strings with delimiters ('-','/','_','+'). The resulting data are converted into a one-hot vector $\mathbf{z}$ and it is fed into a one-hidden layer feed-forward network as $\mathbf{f} = W_z \mathbf{z} + \mathbf{b}_z$.

Entities that co-occur with the target entity also provide clues that can help to infer the type. For example, an entity of the Actor type is likely to co-occur with existing entities of related types such as Film and Award. To obtain entity information, we list entity embeddings $\mathbf{e}_i, i \in [1, E]$ of entities E from the input N posts using the method of Yamada and Shindo [79]. To obtain the relationship

between these entities, we employ self-attention as follows:

$$\alpha_{ij} = \frac{\exp(\sigma(W_u \mathbf{u}_{ij} + \mathbf{b}_u))}{\sum_j \exp(\sigma(W_u \mathbf{u}_{ij} + \mathbf{b}_u))} \tag{5.8}$$

$$\mathbf{u}_{ij} = \tanh(W_e \mathbf{e}_i + W_e \mathbf{e}_j + \mathbf{b}_e) \tag{5.9}$$

$$\acute{\mathbf{e}}_i = \sum_j \alpha_{ij} \mathbf{e}_j \tag{5.10}$$

These representations are concatenated with $\mathbf{f}$ and used as the output $\mathbf{o}_{meta} = [\acute{\mathbf{e}}_i; ...; \acute{\mathbf{e}}_E; \mathbf{f}]$

5.4.2 Context Selection Model

At test time, we input an entity with a burst of posts retrieved by a native string matching. However, those posts can include contexts of homographic entities (*e.g.*, No. 3 for Another Life in Table 5.8) and noisy posts that have no clue on the entity type (*e.g.*, No. 3 for Ben Sheaf in Table 5.8).

To address these issues, we take advantage of emerging contexts of the target entity; if we collect only emerging contexts, 1) we can utilize appropriate contexts for the target entity since two emerging entities with the same name are unlikely to emerge in a short period of time, and 2) emerging contexts by definition include enough information for the readers to understand the target entity.

We thus develop a context selector that predicts whether a given context is an emerging context or not. Specifically, we train a bidirectional GRU, which performs binary classification with the emerging and prevalent contexts collected in § 5.3.2. Using this model, we input each context from the test data and assign a prediction score for the emerging context. For each entity, the top-N contexts of these scores are used as input to the typing model (Figure 5.3).

5.4.3 Model Training

Issues in developing typing and context selection models are how to utilize the constructed training data and how to select the input for the typing model during training. In this study, we simply train each model independently using the same data. Specifically, for the context selection model, we feed the model with

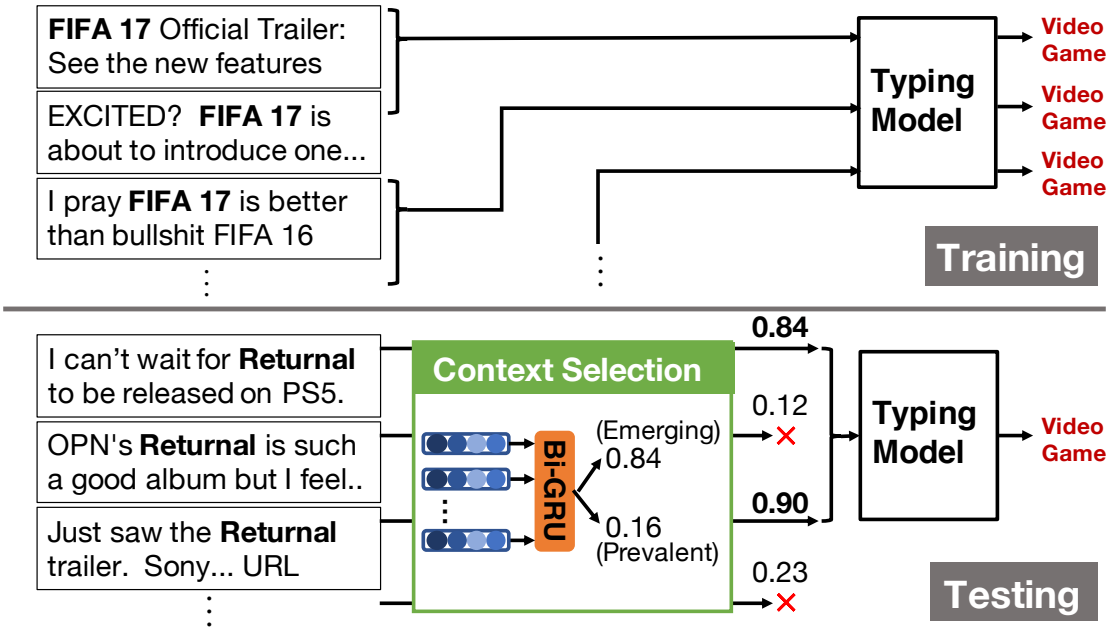

Figure 5.3: Overview of training and testing of the typing model for each entity ($N = 2$). During training, each of the N posts is entered into the model. At test time, top-N posts of the scores obtained by the context selection model are used for prediction.

the emerging and prevalent contexts of the constructed training data. For the typing model, since we use N emerging contexts (posts) during the test time, we repeatedly pick N emerging contexts (posts) in chronological order from the training data and input each N posts into the model to fully exploit a burst of posts of an entity (Figure 5.3).

Here, we perform pretraining with the prevalent contexts and then fine-tune the typing model to improve its robustness. In the experiments, we compare our model with a model that randomly selects contexts for both training and test time.

5.5 Experiments

We performed emerging entity typing using the English and Japanese Twitter datasets built in § 5.3.2.

5.5.1 Models

We describe the typing models compared in the experiments. Since all models employ MI-learning, we use the same parameter N for the models to control the number of input posts.

Proposed (fine-tune) trains the proposed typing model with prevalent contexts, and then performs fine-tuning with emerging contexts. At test time, we applied the context selection model to all the contexts of each entity in the test data to form input.

Proposed (random) randomly extracts 100 contexts per entity from all the collected posts in § 5.3.2 and trains the proposed model. At test time, we randomly selected the contexts for each entity in the test data. This is meant to confirm the effect of discriminating types of contexts (domains).

Yaghoobzadeh uses the model of Yaghoobzadeh et al. [78] modified for our task settings. This model predicts the type of the given entity from its name and contexts using a CNN. Compared to ours, it randomly selects contexts and does not use meta-information. We randomly extracted 100 contexts per entity from the collected contexts in § 5.3.2 and trained the model. At test time, we randomly selected the contexts for each entity in the test data. [5]

5.5.2 Settings

We tokenized each input post using spaCy (ver. 2.0.12)[6] with en_core_web_sm model for English and using MeCab (ver. 0.996)[7] with ipadic (ver. 2.7.0) for Japanese.

We implemented all the models using Keras (ver. 2.3.1).[8] To initialize the word embedding layers for English, we used the 200-dimensional word embeddings pre-trained using GloVe [61] from 2B English posts.[9] For Japanese, we trained 200-dimensional word embeddings using GloVe from 800M Japanese posts

[5] This model also uses a KB embedding of the target entity as a feature. However, since emerging entities in this study are basically absent in KBs, and we cannot acquire those embeddings, we decided not to use them in this comparison.

[6] https://spacy.io

[7] https://taku910.github.io/mecab

[8] https://keras.io

[9] https://nlp.stanford.edu/data/glove.twitter.27B.zip

Parameter	Value
Maximum number of words (Context and CS)	35
Word embedding size (Context, Entity and CS)	200
Dimension of Bi-GRU (Context and CS)	256
Maximum length of entity (Entity)	30
Character embedding size (Entity)	16
Dimension of Bi-GRU (Entity)	64
Maximum number of features (Meta)	20000
Dimension of W_z (Meta)	256
Maximum number of entities (Meta)	$5 * N$
Entity embedding size (Meta)	100
Batch size	32
Dropout	0.5
Adam β_1	0.9
Adam β_2	0.999
Adam ϵ	1e-6

Table 5.5: Hyperparameters of our typing and context selection model. 'Context' means Context Network. 'Entity' means Entity Network. 'Meta' means Meta Network. 'CS' means Context Selection.

posted from Mar. 11th, 2011 to Mar. 11th, 2012 in our Twitter archive. For the Meta Network, from URLs and usernames, we extracted the top 20,000 most frequent tokens in the training data and used them as $\mathbf{z}$ (§ 5.4.1). We used wikipedia2vec[10] with the Wikipedia dump on Dec. 26th, 2015 to extract 100-dimensional embeddings of the entities that cooccur with the target entity.

We optimized all the models using Adam [38]. We finally chose the model at the epoch with the highest accuracy on the development data. We show the detailed hyperparameters of the proposed models in Table 5.5. For the model of Yaghoobzadeh, we adopt the same configurations and hyperparameters of their study.

For each entity in the test data, we perform entity typing once using the selected contexts for each model. For each N, we trained and tested each model 10 times, calculated the micro-F1 [44], and averaged the results.

[10]https://wikipedia2vec.github.io/wikipedia2vec

	All	Person	C. work	Loc.	Group	Event	Device
Proposed (fine-tune)	**0.646**	**0.780**	**0.672**	**0.526**	**0.600**	**0.790**	0.833
Proposed (random)	0.602	0.746	0.629	0.482	0.546	0.780	**0.862**
Yaghoobzadeh	0.582	0.718	0.658	0.348	0.454	0.723	0.824
Majority	N/A	0.145	0.200	0.046	0.156	0.305	0.380

(a) English non-homographic

	All
Proposed (fine-tune)	**0.691**
Proposed (random)	0.579
Yaghoobzadeh	0.575
Majority	N/A

(b) English homographic

	All	Person	C. work	Loc.	Group
Proposed (fine-tune)	**0.766**	**0.822**	**0.870**	**0.729**	**0.846**
Proposed (random)	0.676	0.768	0.790	0.663	0.801
Yaghoobzadeh	0.611	0.675	0.764	0.606	0.729
Majority	N/A	0.095	0.125	0.395	0.840

(c) Japanese non-homographic

	All
Proposed (fine-tune)	**0.665**
Proposed (random)	0.509
Yaghoobzadeh	0.433
Majority	N/A

(d) Japanese homographic

Table 5.6: Micro-F1 for typing emerging entities ($N = 10$). **Majority** predicts the majority label for each type. For homographic entities, we only show the overall results since the number of entities per type is unbalanced.

5.5.3 Results and Analysis

Overall results of the models (N = 10) Table 5.6 shows the results of all types and for each coarse-grained type when $N = 10$. For most of the types, Proposed (fine-tune) outperformed the other methods for both English and Japanese. This indicates the validity of our typing model and the importance of discriminating emerging contexts and others (vs. Proposed (random) and Yaghoobzadeh). Especially for homographic entities, since those entities contain many noisy contexts of other entities, our context selection method that identifies

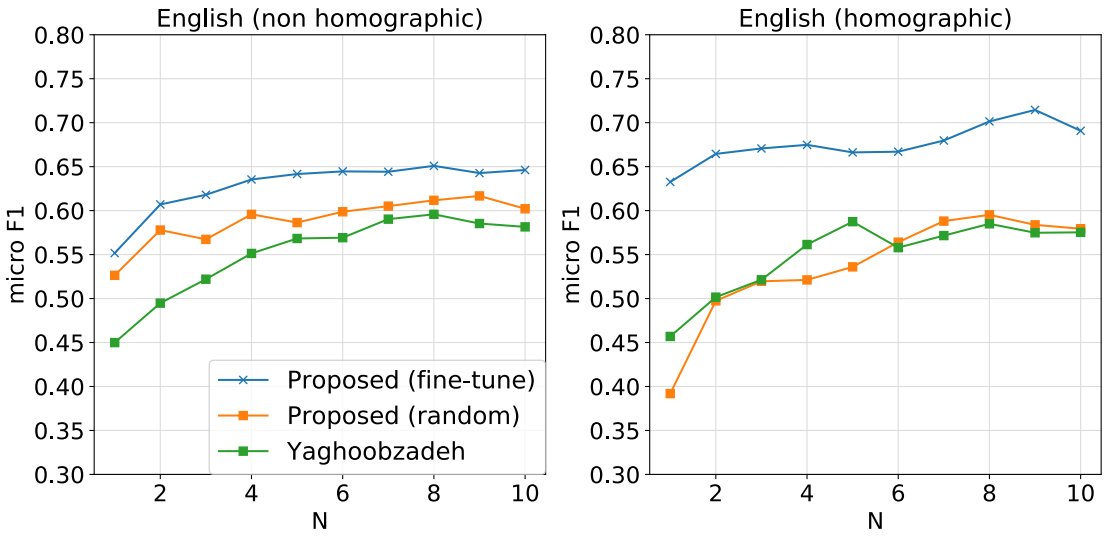

Figure 5.4: Micro-F_1 for each typing model when changing N (English).

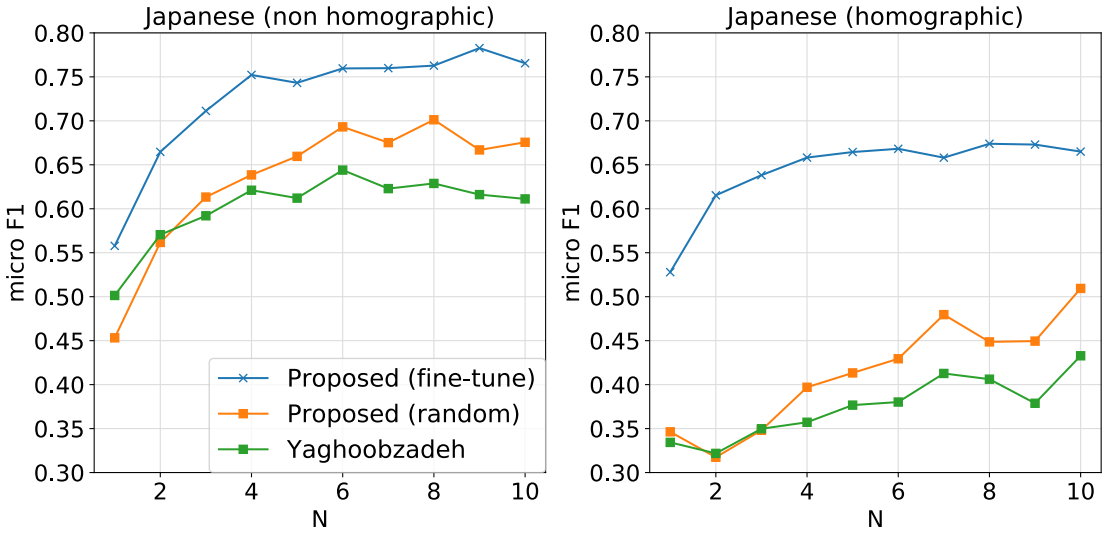

Figure 5.5: Micro-F_1 for each typing model when changing N (Japanese).

the emerging contexts worked effectively. Here, for English, the accuracy of the homographic entities is higher than that of the non-homographic ones. This is probably because the test data of the homographic entities contains more PERSON type entities, which are easier to classify.

Impact of the number of input posts, N Figure 5.4 and 5.5 plot micro F_1 as a function of the number of input posts, N. Although the performances of all the models improve as N is increased, its gain almost converges at $N = 8$. The improvement from $N = 1$ shows the effectiveness of using multiple posts in this task.

Cross-language analysis Interestingly, the performance of Japanese homographic entities is lower than English, even though the number of target types is smaller than that of English (185 vs. 71). This is probably because in languages such as Japanese and Chinese, where entities are not capitalized, their contexts are more likely to be contaminated by common nouns; for example, '香水 (kosui)' refers to both the common noun 'perfume' and the name of the Japanese song released in 2020. In fact in Japanese, the performance of the models without context selection significantly dropped.

Ablation study To verify the contribution of each network of the proposed model, we performed an ablation test. Figure 5.6 and 5.7 show the performance change of Proposed (fine-tune) for each language. We can see that there are significant performance drops when the Context Network is removed. The Entity Network is effective for homographic entities but not for non-homographic entities. Since homographic entities may contain entities with the same name in the training data, it is natural that the Entity Network trained on such data would make biased predictions for such entities. For the Meta Network, it is effective for non-homographic entities with limited contexts ($N < 4$) and homographic entities. Such meta-information helps the model make robust predictions even when the contexts are scarce or contaminated by homographic entities.

Examples Table 5.7 lists examples of non-homographic entities predicted with Proposed (fine-tune). In the first example, although it is difficult to determine its type using only the first context ($N = 1$), by adding another context ($N = 2$), the proposed model utilized it (about a baseball draft) and determined the correct type. The second example is an entity that the proposed model predicted incorrectly. Although we can infer that "Sonos One" is an appliance since it appears with entities like "Google Home" and "Amazon Echo," the proposed method failed to

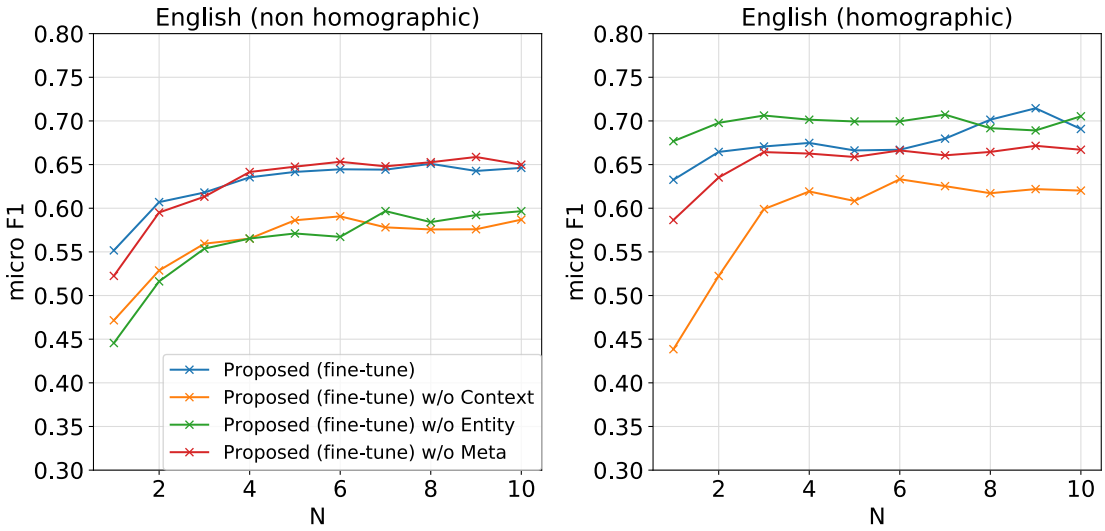

Figure 5.6: Ablation test: micro-F_1 for Proposed (fine-tune) when changing N (English).

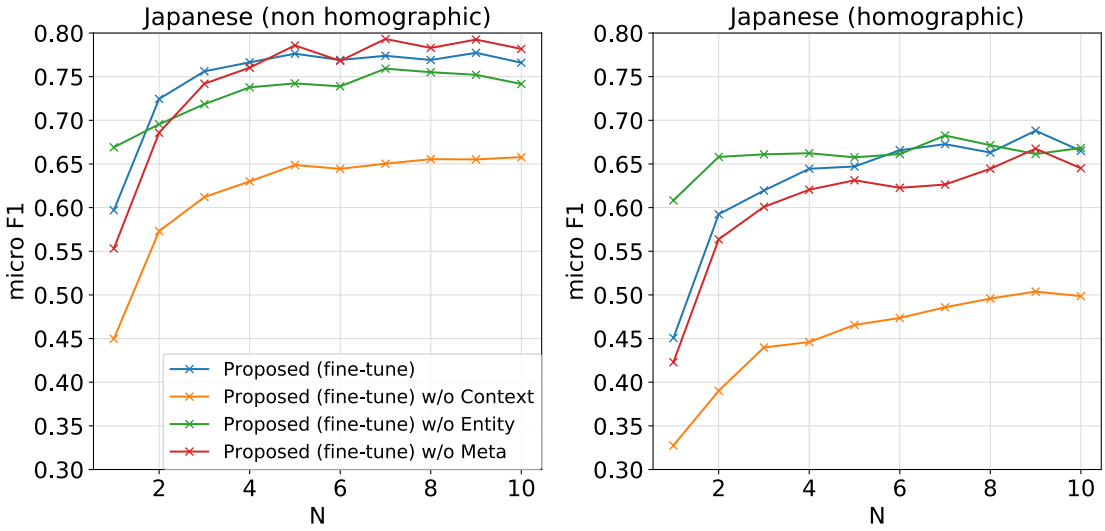

Figure 5.7: Ablation test: micro-F_1 for Proposed (fine-tune) when changing N (Japanese).

Entity: **Tristan Blackmon** True type: BaseballPlayer
1. _USER_'s **Tristan Blackmon** are on the watch list!
2. With the 3rd pick in the 2018 MLS, select **Tristan Blackmon** from the University of the Pacific.

Entity: **Sonos One** True type: InformationAppliance
1. **Sonos One** available on Oct. 24 for $200, preorders starting today. Google assistant coming in 2018 _URL_
2. **Sonos One** is going to combine the best bits from the Amazon Echo and the Google Home: via _URL_

Table 5.7: Examples of **non-homographic** entities that Proposed (fine-tune) predicted correctly (above) and incorrectly (below) (English, $N = 2$).

Entity: **Duck Duck Goose** True type: Film
1. _USER_: Peach and Daisy dance to **Duck Duck Goose** - _USER_ _URL_
2. I'm at play on Broadway and they are playing _USER_ "**Duck Duck Goose**" and I fucking can't. All these mi... _URL_
3. New Trailer: Netflix's "**Duck Duck Goose**" _URL_ _URL_
4. Get ready to play in the **Duck Duck Goose** trailer _URL_

Entity: **Every Day** True type: Book
1. How to Blog **Every Day**: Slides and Resources from WordCamp Portland
2. Solve mysterious questions & Win Mobile Recharge worth Rs.150 **Every Day** Play "Fast & Furious Contest"
3. A weekend Writing Prompt on the Write **Every Day** blog: _URL_ _HASH_ (Thanks to _USER_ for the suggestion!)
4. David Levithan's newest: **Every Day** _URL_ _USER_ _USER_

Table 5.8: Examples of **homographic** entities that Proposed (fine-tune) predicted correctly (above) and incorrectly (below) (English, $N = 4$).

predict the correct type due to the absence of those entities in the period before 2016, when the training data were collected. We thus need to periodically update the training data to cover the latest entities (concepts) using a method like distant supervision.

Table 5.8 lists examples of homographic entities predicted with Proposed (fine-tune). In the first example, our model correctly predicted the type of the target homographic entity by properly selecting the related emerging contexts and utilizing the cooccurring entity "Netflix." For the second example, although our model properly selected the fourth context, it failed to predict due to the lack of clues to identify the type. We might be able to type such entities by using external documents of the URLs in the source posts because we can get more specific contexts from them [62].

5.6 Chapter Summary

We introduced a task of typing emerging entities in microblogs (§ 5.1, 5.2 and 5.3). To perform this task, on the basis of the definition of emerging entities (§ 3.2), we constructed large-scale Twitter datasets for English and Japanese (§ 5.3.2). We developed a modular entity typing model (§ 5.4.1) that encodes different aspects of an emerging entity with MI-learning. To deal with noisy contexts of homographic entities, we adopt a context selection model (§ 5.4.2) that differentiates emerging contexts from others. Experiments (§ 5.5) demonstrated that our method performed more accurately than the baseline model for both non-homographic and homographic emerging entities. We confirmed the importance of selectively using emerging contexts for training and testing the typing model and verified the effectiveness of each network of the proposed typing model.

Chapter 6

Conclusions and Future Work

As discussed in Chapter 1, to monitor microblogs, where useful information is posted in real-time, on an entity basis, it is important to recognize not only existing entities but also emerging and disappearing entities as early as possible. Since it is necessary to properly capture the contexts that represent the emergence or disappearance of these entities, it is difficult to perform recognition relying on language resources such as Wikipedia and Freebase or existing information extraction techniques such as NER, entity linking, and event extraction as they are. In this thesis, we focused on the fact that these contexts are characterized by the timing when they appear and proposed time-sensitive distant supervision that considers the timestamps of microblogs to collect those emerging and disappearing contexts accurately and efficiently. Using the collected contexts, we trained supervised models for entity recognition and typing. With these models, we can detect emerging and disappearing entities appearing in microblogs at an early stage.

The detected entities can be used to maintain a list of monitoring targets in microblog monitoring or to present them to editors as candidates for updating entries in knowledge bases. By performing typing in addition to detection, it is possible to implement applications such as quickly recommending newly appeared or disappeared entities of types that users are interested in for facilitating decision-making. At this time, false positives are an issue in any application, and if not handled properly, they can lead to the spread of serious misinformation such as non-existent events or the death of a living person. As a possible treatment,

instead of presenting only the detected entities and types directly, it is better to provide the source tweets together and let the user decide. Although we used Wikipedia as the knowledge base for building the training and test data in this thesis, a specialized dictionary can be used for the actual applications. For example, instead of Wikipedia entries, we can use lists of restaurant names on sites such as Tabelog[1] to build the training data so that we can concentrate on newly opened or closed restaurants.

In the following sections, we summarize each of the tasks we have done and future works.

6.1 Discovering Emerging Entities in Micloblogs

As a first step, we targeted emerging entities and tried to discover them from microblogs. Those emerging entities are essentially unknown entities due to their emerging nature, and existing studies detected them as out-of-KB entities. This has resulted in the contamination of the training data, as they also target mere out-of-KB entities that are not emerging. To target truly emerging entities, we focused on the characteristic contexts in which they appear and defined these contexts as emerging contexts. We developed time-sensitive distant supervision that considers microblog timestamps to collect emerging contexts efficiently and accurately. In the experiments, we trained a NER model using the collected emerging contexts and applied it to our Twitter archive, and confirmed that our method could detect various emerging entities, including homographic and long-tail ones. Using the entities of Wikipedia, for both English and Japanese, our method found more than 60% of the entities, and most of them can be detected more than a year earlier than their registration in Wikipedia. We plan to improve the method of time-sensitive distant supervision and collect emerging contexts more accurately to remove noise in the training data.

[1]https://tabelog.com/

6.2 Discovering Disappearing Entities in Micloblogs

We next discovered disappearing entities from microblogs. Unlike emerging entities, where the first occurrences of the entity are simply the emerging contexts in most cases, it is difficult to capture the timing of disappearance for disappearing entities. Therefore, if we use time-sensitive distant supervision as-is, a large amount of noise would be included in the resulting training data. Here, we exploited the fact that Wikipedia, the source of entities, lists the year in which entities disappear and collected the posts with the most notable timing in that year as disappearing contexts. In addition, it is difficult to train a recognition model for disappearing entities, which has a small number of entities compared to emerging entities. We focused on the fact that disappearing entities appear in a burst of posts, fine-tuned pretrained word embeddings with those posts, and input them into a NER model to make robust predictions. Experimental results showed that our improved time-sensitive distant supervision collected disappearing contexts properly. The NER model with refined word embeddings using the constructed dataset discovered disappearing entities more accurately than the baseline method, which simply collects the latest burst of posts of the Wikipedia entities as training data. Moreover, our method successfully found more than half of the target disappearing entities in Wikipedia, and some of them were discovered on average one month earlier than the update of the disappearance in Wikipedia. We plan to analyze disappearing entities and their contexts in more detail by type and possibly develop a type-specific method for their recognition.

6.3 Typing Emerging Entities in Micloblogs

We focused on emerging entities, which have no type information registered in the KB as soon as it appears, and performed fine-grained typing of those entities. Compared to existing studies of entity typing that target the text of news articles or knowledge bases, it is difficult to infer the type of emerging entity from a short, noisy post in microblogs. We thus focused on the fact that emerging entities appear as a burst of posts when they first appear and developed a modular neural typing model that encodes these multiple posts using multi-instance learning and

considers meta-information of microblogs in addition to contexts and a surface of the entity. Considering homographic entities, which share the same namings with other entities, the posts in which the entities appear may be contaminated. We, therefore, introduced a context selector that prioritizes emerging contexts, which are most likely to be the posts about the target emerging entity, as the previous stage of the typing model. Experiments demonstrated that our method performed more accurately than the baseline model, which randomly selected input posts and did not include features of meta-information in microblogs for both non-homographic and homographic emerging entities. By focusing on homographic entities, we confirmed the importance of selectively using emerging contexts for training and testing the typing model and verified the effectiveness of each network of the proposed typing model. We plan to perform further profiling of emerging entities such as relation extraction to organize emerging and existing knowledge.

6.4 Other Research Activities in Doctoral Course

Besides the three primary pieces of research conducted in the thesis, I mention the overview of representative studies in my activities.

Conversation Initiation by Diverse News Contents Introduction [3]: In our everyday chit-chat, there is a conversation initiator, who proactively casts an initial utterance to start chatting. Previous studies on conversation systems assumed that the user always initiates conversation and have placed emphasis on how to respond to the given user's utterance. As a result, existing conversation systems become passive; they continue waiting until being spoken to by the users. To this end, we considered the system as a conversation initiator and proposed a novel task of generating the initial utterance in open-domain non-task-oriented conversation. Here, to not make users bored, it is necessary to generate diverse utterances to initiate a conversation without relying on boilerplate utterances like greetings. We thus proposed to generate an initial utterance by summarizing and chatting about news articles, which provide fresh and various contents everyday. To address the lack of training data for this task, we constructed a novel large-scale dataset through crowd-sourcing. To make initial utterances, we presented several

approaches, including information retrieval-based and generation-based models. Experimental results showed that the proposed models trained on our dataset performed reasonably well and outperformed baselines that utilize automatically collected training data in both automatic and manual evaluation.

Chat Detection in an Intelligent Assistant [2]: Recently emerged intelligent assistants on smartphones and home electronics (*e.g.,* Siri and Alexa) can be seen as novel hybrids of domain-specific task-oriented spoken dialogue systems and open-domain non-task-oriented ones. To realize such hybrid dialogue systems, we addressed the task of determining whether or not a user is going to have a chat with the system. To address the lack of benchmark datasets for this task, we constructed a new dataset consisting of utterances collected from the real log data of a commercial intelligent assistant. In addition, we investigated using tweets and Web search queries for handling open-domain user utterances, which characterize the task of chat detection. Experiments demonstrated that, while simple supervised methods are effective, the use of the tweets and search queries further improves the detection accuracy.

Modeling Personal Biases in Language Use by Inducing Personalized Word Embeddings [57, 59, 58]: There exist biases in an individual's language use; the same word (*e.g.,* cool) is used for expressing different meanings (*e.g.,* temperature range) or different words (*e.g.,* cloudy, hazy) are used for describing the same meaning. We proposed a method of modeling such personal biases in word meanings (hereafter, semantic variations) with personalized word embeddings obtained by solving a task on the subjective text while regarding words used by different individuals as different words. To prevent personalized word embeddings from being contaminated by other irrelevant biases, we solved the task of identifying a review-target (objective output) from a given review. To stabilize the training of this extreme multi-class classification, we performed multi-task learning with metadata identification. Experimental results with reviews retrieved from RateBeer confirmed that the obtained personalized word embeddings improved the accuracy of sentiment analysis as well as the target task. Analysis of the obtained personalized word embeddings revealed trends in semantic variations related to frequent and adjective words.

Bibliography

[1] Aguilar, J., Beller, C., McNamee, P., Van Durme, B., Strassel, S., Song, Z., and Ellis, J. (2014). A comparison of the events and relations across ace, ere, tac-kbp, and framenet annotation standards. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 45–53.

[2] Akasaki, S. and Kaji, N. (2017). Chat detection in an intelligent assistant: Combining task-oriented and non-task-oriented spoken dialogue systems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1308–1319.

[3] Akasaki, S. and Kaji, N. (2019). Conversation initiation by diverse news contents introduction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 3988–3998.

[4] Akasaki, S., Yoshinaga, N., and Toyoda, M. (2019). Early discovery of emerging entities in microblogs. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4882–4889.

[5] Akbik, A., Bergmann, T., and Vollgraf, R. (2019). Pooled contextualized embeddings for named entity recognition. In *Proceedings of the 18th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 724–728.

[6] Akbik, A., Blythe, D., and Vollgraf, R. (2018). Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 1638–1649.

[7] Ali, M. A., Sun, Y., Li, B., and Wang, W. (2020). Fine-grained named entity typing over distantly supervised data based on refined representations. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 7391–7398.

[8] Alshaabi, T., Dewhurst, D. R., Minot, J. R., Arnold, M. V., Adams, J. L., Danforth, C. M., and Dodds, P. S. (2021). The growing amplification of social media: measuring temporal and social contagion dynamics for over 150 languages on twitter for 2009–2020. *EPJ Data Science*, 10(1).

[9] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

[10] Augenstein, I., Derczynski, L., and Bontcheva, K. (2017). Generalisation in named entity recognition: A quantitative analysis. *Computer Speech & Language*, 44:61–83.

[11] Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, pages –.

[12] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

[13] Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

[14] Brambilla, M., Ceri, S., Della Valle, E., Volonterio, R., and Acero Salazar, F. X. (2017). Extracting emerging knowledge from social media. In *Proceedings of the 26th International Conference on World Wide Web (WWW)*, pages 795–804.

[15] Bunescu, R. and Paşca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 9–16.

[16] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

[17] Cole-Lewis, H., Pugatch, J., Sanders, A., Varghese, A., Posada, S., Yun, C., Schwarz, M., and Augustson, E. (2015). Social listening: a content analysis of e-cigarette discussions on twitter. *Journal of medical Internet research*, 17(10):e243.

[18] Cowie, J. and Lehnert, W. (1996). Information extraction. *Communications of the ACM*, 39(1):80–91.

[19] Culotta, A. and McCallum, A. (2004). Confidence estimation for information extraction. In *Proceedings of the 5th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 109–112.

[20] Del Corro, L., Abujabal, A., Gemulla, R., and Weikum, G. (2015). Finet: Context-aware fine-grained named entity typing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 868–878.

[21] Derczynski, L., Maynard, D., Aswani, N., and Bontcheva, K. (2013). Microblog-genre noise and impact on semantic annotation accuracy. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media*, pages 21–30.

[22] Derczynski, L., Nichols, E., van Erp, M., and Limsopatham, N. (2017). Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text (WNUT)*, pages 140–147.

[23] Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S. M., and Weischedel, R. M. (2004). The automatic content extraction (ace) program-tasks, data, and evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 837–840.

[24] Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2):179–211.

[25] Färber, M., Rettinger, A., and Asmar, B. (2016). On emerging entity detection. In *Proceedings of the 20th International Conference on Knowledge Engineering and Knowledge Management (EKAW)*, pages 223–238.

[26] Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*.

[27] Fleiss, J. L. and Cohen, J. (1973). The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

[28] Forney, G. D. (1973). The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

[29] Fukuda, N., Yoshinaga, N., and Kitsuregawa, M. (2020). Robust backed-off estimation of out-of-vocabulary embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings (EMNLP-Fingings)*, pages 4827–4838.

[30] Graus, D., Odijk, D., and de Rijke, M. (2018). The birth of collective memories: Analyzing emerging entities in text streams. *Journal of the Association for Information Science and Technology*, 69(6):773–786.

[31] Han, J., Pei, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., and Hsu, M. (2001). Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceedings of the 17th international conference on data engineering (ICDE)*, pages 215–224. Citeseer.

[32] Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116.

[33] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[34] Hoffart, J., Altun, Y., and Weikum, G. (2014). Discovering emerging entities with ambiguous names. In *Proceedings of the 23rd International Conference on World Wide Web (WWW)*, pages 385–396.

[35] Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

[36] Ji, H., Grishman, R., Dang, H. T., Griffitt, K., and Ellis, J. (2010). Overview of the tac 2010 knowledge base population track. In *Third text analysis conference (TAC 2010)*, volume 3, pages 3–3.

[37] Kim, S. K., Park, M. J., and Rho, J. J. (2015). Effect of the government's use of social media on the reliability of the government: Focus on twitter. *Public Management Review*, 17(3):328–355.

[38] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, pages –.

[39] Kolitsas, N., Ganea, O.-E., and Hofmann, T. (2018). End-to-end neural entity linking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning (CoNLL)*, pages 519–529.

[40] Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning (ICML*, pages 282–289.

[41] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 260–270.

[42] Lin, T., Etzioni, O., et al. (2012). No noun phrase left behind: detecting and typing unlinkable entities. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 893–903.

[43] Lin, Z., Feng, M., Santos, C. N. d., Yu, M., Xiang, B., Zhou, B., and Bengio, Y. (2017). A structured self-attentive sentence embedding. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, pages –.

[44] Ling, X. and Weld, D. S. (2012). Fine-grained entity recognition. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pages 94–100.

[45] Liu, J., Chen, Y., Liu, K., Bi, W., and Liu, X. (2020). Event extraction as machine reading comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1641–1651.

[46] Lu, J. and Ng, V. (2017). Joint learning for event coreference resolution. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 90–101.

[47] Ma, X. and Hovy, E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1064–1074.

[48] Mai, K., Pham, T.-H., Nguyen, M. T., Nguyen, T. D., Bollegala, D., Sasano, R., and Sekine, S. (2018). An empirical study on fine-grained named entity recognition. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 711–722.

[49] Martins, P. H., Marinho, Z., and Martins, A. F. (2019). Joint learning of named entity recognition and entity linking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop (ACL-SW)*, pages 190–196.

[50] McClosky, D. and Manning, C. D. (2012). Learning constraints for consistent timeline extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 873–882.

[51] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS)*, pages 3111–3119.

[52] Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 1003–1011.

[53] Nadeau, D. and Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

[54] Nakashole, N., Tylenda, T., and Weikum, G. (2013). Fine-grained semantic typing of emerging entities. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1488–1497.

[55] Navigli, R. (2009). Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, 41(2):1–69.

[56] Nguyen, T. H. and Grishman, R. (2015). Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 365–371.

[57] Oba, D., Sato, S., Akasaki, S., Yoshinaga, N., and Toyoda, M. (2020). Personal semantic variations in word meanings: Induction, application, and analysis. *Journal of Natural Language Processing*, 27(2):467–490.

[58] Oba, D., Sato, S., Yoshinaga, N., Akasaki, S., and Toyoda, M. (2019a). Understanding interpersonal variations in word meanings via review target identification. In *Proceedings of the 20th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, number 129.

[59] Oba, D., Yoshinaga, N., Sato, S., Akasaki, S., and Toyoda, M. (2019b). Modeling personal biases in language use by inducing personalized word embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2102–2108.

[60] Obeidat, R., Fern, X., Shahbazi, H., and Tadepalli, P. (2019). Description-based zero-shot fine-grained entity typing. In *Proceedings of the 18th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (NAACL-HLT)*, pages 807–814.

[61] Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

[62] Plank, B., Hovy, D., McDonald, R., and Søgaard, A. (2014). Adapting taggers to twitter with not-so-distant supervision. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING)*, pages 1783–1792.

[63] Ratinov, L. and Roth, D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL)*, pages 147–155.

[64] Riedel, S., Yao, L., and McCallum, A. (2010). Modeling relations and their mentions without labeled text. In *Proceedings of the 2010 European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD)*, pages 148–163.

[65] Ritter, A., Clark, S., Etzioni, O., et al. (2011). Named entity recognition in tweets: an experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1524–1534.

[66] Ritter, A., Etzioni, O., and Clark, S. (2012). Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, pages 1104–1112.

[67] Sang, E. T. K. and De Meulder, F. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the 7th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (NAACL-HLT)*, pages 142–147.

[68] Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

[69] Shen, W., Wang, J., and Han, J. (2014). Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, 27(2):443–460.

[70] Shimaoka, S., Stenetorp, P., Inui, K., and Riedel, S. (2017). Neural architectures for fine-grained entity type classification. In *Proceedings of the 15th conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 1271–1280.

[71] Wang, X., Jiang, Y., Bach, N., Wang, T., Huang, Z., Huang, F., and Tu, K. (2021a). Automated concatenation of embeddings for structured prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 2643–2660.

[72] Wang, X., Jiang, Y., Bach, N., Wang, T., Huang, Z., Huang, F., and Tu, K. (2021b). Improving named entity recognition by external context retrieving and cooperative learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 1800–1812.

[73] Weikum, G., Dong, L., Razniewski, S., and Suchanek, F. (2020). Machine knowledge: Creation and curation of comprehensive knowledge bases. *arXiv preprint arXiv:2009.11564*.

[74] Weischedel, R., Palmer, M., Marcus, M., Hovy, E., Pradhan, S., Ramshaw, L., Xue, N., Taylor, A., Kaufman, J., Franchini, M., et al. (2013). Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23.

[75] Wu, Z., Song, Y., and Giles, C. L. (2016). Exploring multiple feature spaces for novel entity discovery. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 3073–3079.

[76] Xin, J., Lin, Y., Liu, Z., and Sun, M. (2018). Improving neural fine-grained entity typing with knowledge attention. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*.

[77] Xu, B., Luo, Z., Huang, L., Liang, B., Xiao, Y., Yang, D., and Wang, W. (2018). Metic: Multi-instance entity typing from corpus. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 903–912.

[78] Yaghoobzadeh, Y., Adel, H., and Schütze, H. (2018). Corpus-level fine-grained entity typing. *Journal of Artificial Intelligence Research*, 61:835–862.

[79] Yamada, I. and Shindo, H. (2019). Neural attentive bag-of-entities model for text classification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 563–573.

[80] Yang, J., Liang, S., and Zhang, Y. (2018). Design challenges and misconceptions in neural sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 3879–3889.

[81] Yin, W., Kann, K., Yu, M., and Schütze, H. (2017). Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*.

[82] Zhang, S., Gosselt, J. F., and de Jong, M. D. T. (2020). How large information technology companies use twitter: Arrangement of corporate accounts and characteristics of tweets. *Journal of Business and Technical Communication*, 34(4):364–392.

Publications

Publications related to the thesis

International conference (reviewed)

[1] Satoshi Akasaki, Naoki Yoshinaga, Masashi Toyoda. “Fine-grained Typing of Emerging Entities in Microblogs.” (EMNLP 2021, findings).

[2] Satoshi Akasaki, Naoki Yoshinaga, Masashi Toyoda. “Early Discovery of Emerging Entities in Microblogs.” (IJCAI 2019).

Domestic conference

[3] 赤﨑智，吉永直樹，豊田正史. “マイクロブログからの消失エンティティの検知”. 言語処理学会第28回年次大会 (NLP 2022).

[4] 赤﨑智，吉永直樹，豊田正史. “ソーシャルメディアストリームからの新固有表現の発見”. 第32回人工知能学会全国大会 (JSAI 2018).

Publications non-related to the thesis

Journal

[5] Daisuke Oba, Shoetsu Sato, Satoshi Akasaki, Naoki Yoshinaga, Masashi Toyoda. “Personal Semantic Variations in Word Meanings: Induction, Appli-

cation, and Analysis." Journal of Natural Language Processsing, Volume 27, Number 2, June 2020

International conference

[6] Satoshi Akasaki, Nobuhiro Kaji. "Conversation Initiation by Diverse News Contents Introduction." (NAACL 2019).

[7] Daisuke Oba, Naoki Yoshinaga, Shoetsu Sato, Satoshi Akasaki and Masashi Toyoda. "Modeling Personal Biases in Language Use by Inducing Personalized Word Embeddings." (NAACL 2019)

[8] Daisuke Oba, Shoetsu Sato, Naoki Yoshinaga, Satoshi Akasaki, Masashi Toyoda. "Understanding Interpersonal Variations in Word Meanings via Review Target Identification." Proceedings of the 20th International Conference on Computational Linguistics and Lntelligent Text Processing (CICLing 2019)

[9] Satoshi Akasaki, Nobuhiro Kaji. "Chat Detection in an Intelligent Assistant: Combining Task-oriented and Non-task-oriented Spoken Dialogue Systems." (ACL 2017).

Domestic conference

[10] 叶内晨，赤﨑智，堀江伸太朗，小亀俊太郎. "Capex 雑談対話コーパスの構築とその分析". 言語処理学会第28回年次大会 (NLP 2022).

[11] 大葉大輔，佐藤翔悦，赤﨑智，吉永直樹，豊田正史. "人の言語使用における単語の意味の揺らぎの解明に向けて". 言語処理学会第25回年次大会 (NLP 2019)

[12] 赤﨑智，鍜治伸裕. "知的対話アシスタントにおける雑談を目的としたユーザ発話の検出". 第231回NL・第116回SLP合同研究発表会 (SIG-NL/SIG-SLP)

[13] 赤﨑智，鍜治伸裕. "知的対話アシスタントにおける発話の雑談意図の判定". 言語処理学会第23回年次大会 (NLP 2017)

[14] 赤﨑智，吉永直樹，豊田正史. "発生普及過程を捉えた未知エンティティの発見". 第9回データ工学と情報マネジメントに関するフォーラム (DEIM 2017).