Fine-grained Typing of Emerging Entities in Microblogs

Satoshi Akasaki
The University of Tokyo

Naoki Yoshinaga
Institute of Industrial Science, the University of Tokyo

Masashi Toyoda
Institute of Industrial Science, The University of Tokyo

{akasaki,ynaga,toyoda}@tkl.iis.u-tokyo.ac.jp

Abstract

Analyzing microblogs where we post what we experience enables us to perform various applications such as social-trend analysis and entity recommendation. To track emerging trends in a variety of areas, we want to categorize information on emerging entities (*e.g.,* Avatar 2) in microblog posts according to their types (*e.g.,* Film). We thus introduce a new entity typing task that assigns a fine-grained type to each emerging entity when a burst of posts containing that entity is first observed in a microblog. The challenge is to perform typing from noisy microblog posts without relying on prior knowledge of the target entity. To tackle this task, we build large-scale Twitter datasets for English and Japanese using time-sensitive distant supervision. We then propose a modular neural typing model that encodes not only the entity and its contexts but also meta information in multiple posts. To type 'homographic' emerging entities (*e.g.,* 'Go' means an emerging programming language and a classic board game), which contexts are noisy, we devise a context selector that finds related contexts of the target entity. Experiments on the Twitter datasets confirm the effectiveness of our typing model and the context selector.

1 Introduction

Microblogs enable us to instantly share a wider variety of topics than news streams (Graus et al., 2018) and have become one of the primary sources for acquiring new information. To analyze this huge 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 (Weikum et al., 2020). 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.

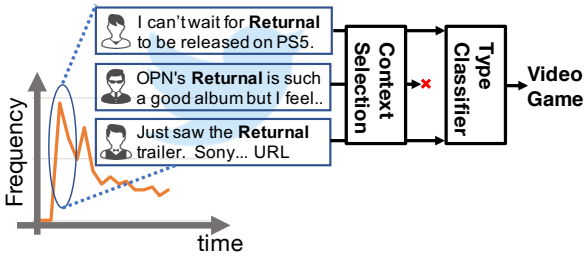

Figure 1: Emerging entity typing: identify the type of a given emerging entity with its first burst of posts.

Motivated by this background, Akasaki et al. (2019) (§ 2.1) defined emerging entities as that appear in contexts that emphasize their novelty, and attempted to discover emerging entities from microblogs. To extract emerging entities, they 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 1)) (§ 3.1), and developed a method of discovering them from microblogs. Although their method detected emerging entities promptly, typing those emerging entities is still necessary for usage in the downstream applications.

Existing studies on entity typing, however, focus on non-emerging (or prevalent) entities (Ling and Weld, 2012; Shimaoka et al., 2017; Xin et al., 2018; Obeidat et al., 2019; Ali et al., 2020) (§ 2.2). Most of them classify single mentions of entities into their context-dependent types. To complement a scarce context, many studies 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 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 1, § 3.2), assuming that the target mention is detected in advance. This is a more realistic setting for typing emerging entities

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 4667–4679
November 7–11, 2021.

than the conventional mention-level typing.

To build training data for this task (§ 3.3), we collect emerging entities and their contexts for English and Japanese using distant supervision (Mintz et al., 2009; Akasaki et al., 2019). 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 (Riedel et al., 2010; Yaghoobzadeh et al., 2018) (§ 4.1). In addition to contexts for the entity and its entity surface, this model leverages meta-information such as URLs and usernames, 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 (§ 4.2).

We finally evaluate our typing model on the above English and Japanese Twitter datasets (§ 5). 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 (§ 3.2).
- We built two large-scale Twitter datasets for English and Japanese (§ 3.3). We will release them to facilitate future studies.
- We proposed an entity typing model (§ 4.1) and a context selection model (§ 4.2) 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 the emerging entities. We then explain the existing task settings of entity typing and discuss their limitations.

2.1 Emerging Entity Detection

Although there are studies that find "emerging" entities (Nakashole et al., 2013; Hoffart et al., 2014; Wu et al., 2016; Derczynski et al., 2017), 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, Akasaki et al. (2019) defined emerging entities as those which appear in emerging contexts that emphasize their novelty (§ 3.1). With this definition, they developed a method called time-sensitive distant supervision, which uses time-stamps of microblogs to collect early posts (contexts) in which 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 proposed by Akasaki et al. (2019) and conduct time-sensitive distant supervision to automatically construct large-scale English and Japanese Twitter datasets for typing emerging entities.

2.2 Entity Typing

Traditionally, named entity recognition (Sang and De Meulder, 2003; Ritter et al., 2011; Weischedel et al., 2013; Ma and Hovy, 2016; Akbik et al., 2019) jointly performs recognition and typing of entity mentions in the text. However, most of the 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) (Mai et al., 2018).

Focusing on fine-grained entity typing, recent studies adopted distant supervision (Mintz et al., 2009) that automatically annotates entities with KB categories, and tackled the task of classifying single mentions of entities with their types in a context (Ling and Weld, 2012). This allows us to exploit resource-hungry neural models (Shimaoka et al., 2017) and knowledge of the target entity derived from KBs (Obeidat et al., 2019; Xin et al., 2018) or a large corpus (Del Corro et al., 2015). 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. (2012) and Nakashole et al. (2013) 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. (2018) and Xu et al. (2018) 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, in order 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 (§ 3.2). We build Twitter datasets for this task (§ 3.3) and develop an effective typing method (§ 4).

3 Task and Datasets

This section first introduces the definition of emerging entities (Akasaki et al., 2019) and then defines our task of typing emerging entities. Finally, we describe our dataset for this task.

3.1 Definition of Emerging Entity

We adopt the same definition of emerging entity as in Akasaki et al. (2019) to focus on truly emerging entities. They defined emerging entities as follows, inspired by the fact that microblog users mention emerging entities that are not yet well known in characteristic contexts (*emerging contexts*):

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.*

They built a Japanese dataset of emerging entities with emerging contexts by collecting early time-stamped posts of Wikipedia entities from Twitter by using time-sensitive distant supervision. Since their dataset does not include type information, we reconstruct it with types in English and Japanese from scratch.

3.2 Task Settings

Inspired by the related studies on entity typing (§ 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 typing. In fact, the accuracy of Twitter NER is very low (29.7%) for out-of-vocabulary entities (Fukuda et al., 2020). 3) Emerging entities show an early burst of posts around the time of their introduction into public discourse (Graus et al., 2018). These considerations lead us to the following task settings:

Fine-grained emerging entity typing. *Given an entity and a burst of posts containing the entity, the goal of the task is to predict the single type of the 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.

3.3 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 (Alshaabi et al., 2021). 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

[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

the actual data (Alshaabi et al., 2021)). 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.

3.3.1 Training Data

To create the training data, we used time-sensitive distant supervision (Akasaki et al., 2019) 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 that were 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 for pretraining the typing model.

We mapped the collected entities to their corresponding fine-grained types assigned in the DBpedia (Auer et al., 2007) 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 Akasaki et al. (2019). 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 440,751 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.

TYPE DBpedia types	#ent.	#posts
PERSON	**9878**	**316123**
Person (Misc.)	2514	73517
SoccerPlayer	1337	41955
(A)FootballPlayer	1157	43737
Others (70 types)	4870	156914
CREATIVEWORK	**6979**	**192214**
Film	1777	46185
Album	1272	31947
TelevisionShow	1043	26526
Others (22 types)	2887	87556
LOCATION	**1588**	**31554**
City	912	14566
Building	146	3922
Stadium	66	2260
Others (33 types)	464	10806
GROUP	**1413**	**39260**
Company	719	20148
Organisation	223	6172
Others (21 types)	471	12940
EVENT	**378**	**9014**
Award	110	2593
SpaceMission	46	910
Others (18 types)	222	5511
DEVICE	**335**	**9404**
Device	147	4053
Automobile	69	2100
Others (6 types)	119	3251
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
Others (24 types)	1897	52762
CREATIVEWORK	**5706**	**140191**
Single	1211	28985
TelevisionShow	1058	26488
Album	842	18436
Film	799	20075
Others (10 types)	1796	46207
LOCATION	**304**	**6419**
Building	98	2421
Museum	33	775
Station	32	694
Settlement	23	376
Others (21 types)	118	2153
GROUP	**310**	**7667**
Company	216	5373
SoccerClub	48	1075
Organisation	24	642
PoliticalParty	22	577
TOTAL	**10315**	**259484**

(b) Japanese data

Table 1: Statistics of emerging entities and a burst of posts in the training data obtained from Twitter.

Table 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.

3.3.2 Test Data

For non-homographic emerging entities, we built the test data in a similar way as 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-

[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).

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

rater agreement of 0.782 for English and 0.771 for Japanese by Fleiss' Kappa (Fleiss and Cohen, 1973); both show substantial agreement. We finally obtained 31,450 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 (see Appendix (Table 5) for the statistics).

For homographic emerging entities, we manually constructed the test data since it is difficult to collect their contexts using distant supervision. We collected the titles of Wikipedia articles, each of which has a disambiguation page, and gather 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 for 200 entities in Japanese (see Appendix (Table 6) for the statistics).

Table 2 shows some examples of collected entities and their posts (excerpts). The first example is a non-homographic emerging entity in the training

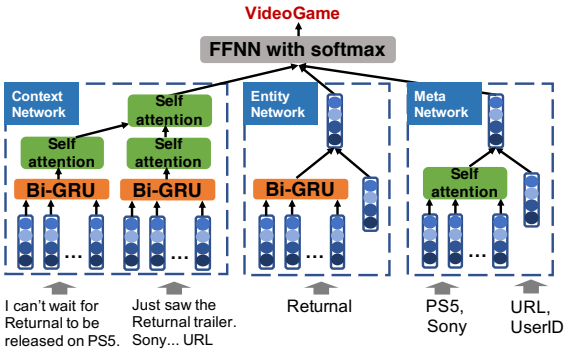

Figure 2: Overview of our entity typing model ($N = 2$): three networks process contexts, entity, and meta-information, respectively using MI-learning.

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 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 (§ 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 (§ 4.2). In the following, we describe the details of each model and how to train and test the models.

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 2): Context Network and Entity Network that encode contexts and entities, which are based on Yaghoobzadeh et al. (2018) while refining their classic CNN-based structure with

GRU (Bahdanau et al., 2015) and self-attention mechanism (Lin et al., 2017), and Meta Network that encodes meta-information specific to microblogs. We rely on MI-learning (Riedel et al., 2010), 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 $p = \text{softmax}(W_o[o_{context}; o_{entity}; o_{meta}] + b_o)$. We describe the details of each network hereafter.

4.1.1 Context Network

This model captures contexts of given posts; it differs from the Context Model (Yaghoobzadeh et al., 2018) in that we change CNN to GRU and introduce a self-attention mechanism to capture longer relationships and dependencies between words (Yin et al., 2017). Specifically, we encode the given entity using MI-learning by inputting N contexts where the entity appears. We convert each word $w_{it}, t \in [1, S]$ of the i-th input context to x_{it} using the embedding matrix W_w, $x_{it} = W_w w_{it}$. We input this into a bi-directional GRU as $h_{it} = \text{BiGRU}(x_{it})$, and apply self-attention to the entire hidden states to capture the word relations:

$$\alpha_{ijk} = \frac{\exp(\sigma(W_u u_{ijk} + b_u))}{\sum_k \exp(\sigma(W_u u_{ijk} + b_u))} \quad (1)$$

$$u_{ijk} = \tanh(W_h h_{ij} + W_h h_{ik} + b_h) \quad (2)$$

$$\acute{h}_{ij} = \sum_k \alpha_{ijk} h_{ik} \quad (3)$$

We first obtain the similarity u_{ijk} between h_{ij} and 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 $\acute{h}_{ij}$ as a weighted sum of the hidden layers. These $\acute{h}_{ij}$ are concatenated to form the sentence representation $s_i = [\acute{h}_{i1}; ...; \acute{h}_{iS}]$.

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

$$\alpha_{ij} = \frac{\exp(\sigma(W_u u_{ij} + b_u))}{\sum_j \exp(\sigma(W_u u_{ij} + b_u))} \quad (4)$$

$$u_{ij} = \tanh(W_s s_i + W_s s_j + b_s) \quad (5)$$

$$\acute{s}_i = \sum_j \alpha_{ij} s_j \quad (6)$$

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

4.1.2 Entity Network

This model captures a given entity surface; it differs from the Global Model (Yaghoobzadeh et al., 2018), 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 $c_i, t \in [1, C]$ of the target entity to x_i using the embedding matrix W_c, $x_i = W_c c_i$. Similarly to the Context Network, we input this into a bi-directional GRU and obtain the character-based entity representation as $h = \text{BiGRU}(x_i)$.

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

4.1.3 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 z and it is fed into a one-hidden layer feed-forward network as $f = W_z z + b_z$.

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 $e_i, i \in [1, E]$ from the input N posts using the method of Yamada and Shindo (2019). To obtain the relationship between

these entities, we employ self-attention as follows:

$$\alpha_{ij} = \frac{\exp(\sigma(W_u u_{ij} + b_u))}{\sum_j \exp(\sigma(W_u u_{ij} + b_u))} \quad (7)$$

$$u_{ij} = \tanh(W_e e_i + W_e e_j + b_x) \quad (8)$$

$$\acute{e}_i = \sum_j \alpha_{ij} e_j \quad (9)$$

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

4.2 Context Selection Model

At test time, we input an entity with a burst of posts, which are retrieved by a native string matching. However, those posts can include contexts of homographic entities (*e.g.*, No. 3 for Another Life in Table 2) and noisy posts that have no clue on the entity type (*e.g.,* No. 3 for Ben Sheaf in Table 2).

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 bi-directional GRU, which performs binary classification with the emerging and prevalent contexts collected in § 3.3. 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 3).

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 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 3).

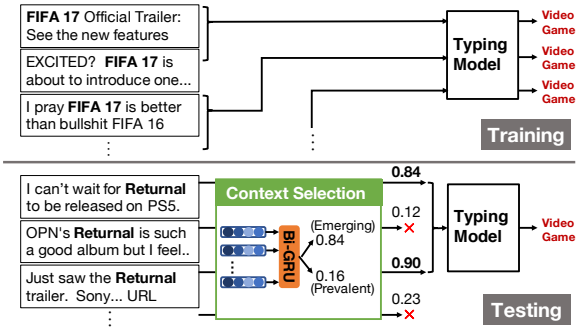

Figure 3: Overview of training and testing of the typing model for each entity ($N = 2$). During training, each of the N posts is entered into the model. At test time, top-N posts of the scores obtained by the context selection model are used for prediction.

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 Experiments

We performed emerging entity typing using the English and Japanese Twitter datasets built in § 3.3.

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 § 3.3 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. (2018) 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 § 3.3 and trained the model. At test time, we randomly selected the contexts for each entity in the test data.

	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 3: 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.

5.2 Settings

We tokenized each input post using spaCy (ver. 2.0.12)[4] with en_core_web_sm model for English and using MeCab (ver. 0.996)[5] with ipadic (ver. 2.7.0) for Japanese.

We implemented all the models using Keras (ver. 2.3.1).[6] To initialize the word embedding layers for English, we used the 200-dimensional word embeddings pre-trained using GloVe (Pennington et al., 2014) from 2B English posts.[7] For Japanese, we trained 200-dimensional word embeddings using GloVe from 800M Japanese posts 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 as z (§ 4.1.3). We used wikipedia2vec[8] 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 (Kingma and Ba, 2015). We finally chose the model at the epoch with the highest accuracy on the development data. We show the detailed hyperparameters of the models in Appendix (Table 7). 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 (Ling and Weld, 2012), and averaged the results.

[4] https://spacy.io
[5] https://taku910.github.io/mecab
[6] https://keras.io
[7] https://nlp.stanford.edu/data/glove.twitter.27B.zip
[8] https://wikipedia2vec.github.io/wikipedia2vec

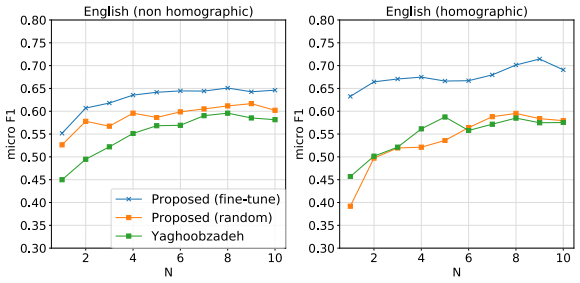

Figure 4: Micro-F_1 for each typing model when changing N (English).

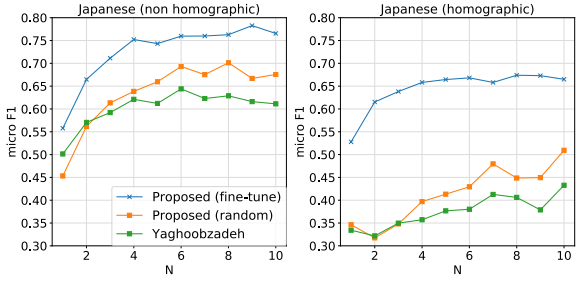

Figure 5: Micro-F_1 for each typing model when changing N (Japanese).

5.3 Results and Analysis

Table 3 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)). Especially for homographic entities, since those entities contain many noisy contexts of other entities, our context selection method that identifies the emerging contexts worked effectively.

Impact of the number of input posts, N Figure 4 and 5 plot micro F_1 as a function of the num-

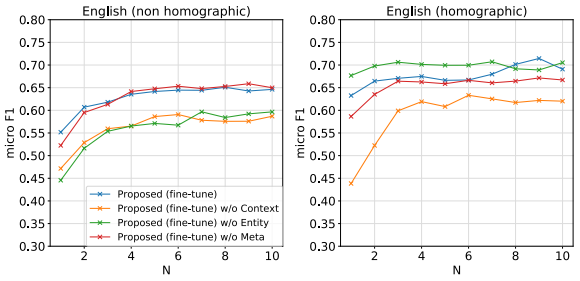

Figure 6: Ablation test: micro-F_1 for Proposed (fine-tune) when changing N (English)

Entity: **Tristan Blackmon** 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** Type: Appliance
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 4: Examples that our model predicted correctly (above) and incorrectly (below) (English, $N = 2$)

ber 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 6 shows the performance change of Proposed (fine-tune) for the English data. 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 4 lists examples of predictions 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 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 update the training data periodically to cover the latest entities (concepts) by using a method like distant supervision.

6 Conclusions

We introduced a task of typing emerging entities in microblogs (§ 3.2). To perform this task, on the basis of the definition of emerging entities (§ 3.1), we constructed large-scale Twitter datasets for English and Japanese (§ 3.3). We developed a modular entity typing model (§ 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 (§ 4.2) that differentiates emerging contexts from others. Experiments (§ 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.

For future work, we plan to perform further profiling of emerging entities such as relation extraction to organize emerging and existing knowledge. We release the dataset used in our experiments.[9]

[9] http://www.tkl.iis.u-tokyo.ac.jp/~akasaki/emnlp21.html

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 18J22830 and 21H03494. We thank Joshua Tanner for his help in writing the paper. We also thank the volunteers and the anonymous reviewers for their hard work.

Ethical Considerations

We collected the dataset (§ 3.3) through the official Twitter APIs so that it conforms to Twitter's terms of service. We release only the Tweet IDs of the tweets used in the experiments and we confirmed that their redistribution is in accordance with Twitter's developer policies.[10] Researchers cannot collect deleted tweets or tweets of private users, which protects users' privacy.

References

Satoshi Akasaki, Naoki Yoshinaga, and Masashi Toyoda. 2019. Early discovery of emerging entities in microblogs. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4882–4889.

Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 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.

Muhammad Asif Ali, Yifang Sun, Bing Li, and Wei Wang. 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.

Thayer Alshaabi, David Rushing Dewhurst, Joshua R. Minot, Michael V. Arnold, Jane L. Adams, Christopher M. Danforth, and Peter Sheridan Dodds. 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).

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, pages –.

[10] https://developer.twitter.com/en/developer-terms/agreement-and-policy

Luciano Del Corro, Abdalghani Abujabal, Rainer Gemulla, and Gerhard Weikum. 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.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 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.

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

Nobukazu Fukuda, Naoki Yoshinaga, and Masaru Kitsuregawa. 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.

David Graus, Daan Odijk, and Maarten de Rijke. 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.

Johannes Hoffart, Yasemin Altun, and Gerhard Weikum. 2014. Discovering emerging entities with ambiguous names. In *Proceedings of the 23rd International Conference on World Wide Web (WWW)*, pages 385–396.

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

Thomas Lin, Oren Etzioni, 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.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, pages –.

Xiao Ling and Daniel S Weld. 2012. Fine-grained entity recognition. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pages 94–100.

Xuezhe Ma and Eduard Hovy. 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.

Khai Mai, Thai-Hoang Pham, Minh Trung Nguyen, Tuan Duc Nguyen, Danushka Bollegala, Ryohei Sasano, and Satoshi Sekine. 2018. An empirical study on fine-grained named entity recognition. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 711–722.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 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.

Ndapandula Nakashole, Tomasz Tylenda, and Gerhard Weikum. 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.

Rasha Obeidat, Xiaoli Fern, Hamed Shahbazi, and Prasad Tadepalli. 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.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 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.

Alan Ritter, Sam Clark, Oren Etzioni, 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.

Erik Tjong Kim Sang and Fien De Meulder. 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.

Sonse Shimaoka, Pontus Stenetorp, Kentaro Inui, and Sebastian Riedel. 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.

Gerhard Weikum, Luna Dong, Simon Razniewski, and Fabian Suchanek. 2020. Machine knowledge: Creation and curation of comprehensive knowledge bases. *arXiv preprint arXiv:2009.11564*.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23.

Zhaohui Wu, Yang Song, and C Lee Giles. 2016. Exploring multiple feature spaces for novel entity discovery. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 3073–3079.

Ji Xin, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2018. Improving neural fine-grained entity typing with knowledge attention. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*.

Bo Xu, Zheng Luo, Luyang Huang, Bin Liang, Yanghua Xiao, Deqing Yang, and Wei Wang. 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.

Yadollah Yaghoobzadeh, Heike Adel, and Hinrich Schütze. 2018. Corpus-level fine-grained entity typing. *Journal of Artificial Intelligence Research*, 61:835–862.

Ikuya Yamada and Hiroyuki Shindo. 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.

Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. 2017. Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*.

A Appendix

A.1 Annotation Guideline

We provided the following instructions and some examples (*e.g.,* Table 2) for annotators. These instructions are translated from Japanese to promote readability.

1. Please read the following definition of emerging entities (omitted here since it is identical to the one given in § 3.1) carefully and see examples of emerging entities.

2. For non-homographic entities, you will be given entities and their tweets. Please check these tweets and then label the entity as "emerging" if one or more emerging contexts for the entity appear.

3. For homographic entities, you will be given tweets with entities on each day. Please check these tweets in date order and "fill in the dates" when one or more emerging contexts for the entity appear.

A.2 Data Statistics

Table 5 and 6 show the statistics of the test data. As for homographic entities, the data is unbalanced because they tended to be concentrated in certain types, such as names of people and creative works.

A.3 Hyperparameters

Table 7 shows the hyperparameters of our typing model and context selection model.

TYPE DBpedia types	#ent.	#posts
PERSON	**200**	**6048**
SoccerPlayer	51	1447
Politician	36	875
Person (Misc.)	29	839
(A)FootballPlayer	15	518
Others (23 types)	69	2369
CREATIVEWORK	**200**	**5327**
Album	50	1280
Film	40	1097
TelevisionShow	37	750
VideoGame	28	948
Others (12 types)	45	1252
LOCATION	**200**	**5687**
Stadium	38	980
Building	35	1337
Museum	19	424
Station	15	554
Others (25 types)	93	2392
GROUP	**200**	**4907**
Organisation	44	1077
PoliticalParty	36	808
Company	33	942
SoccerClub	18	407
Others (12 types)	69	1673
EVENT	**200**	**3973**
Award	61	1064
GrandPrix	21	443
WrestlingEvent	14	340
MMA Event	12	261
Others (14 types)	92	1865
DEVICE	**200**	**5302**
Device	76	2070
Automobile	45	1343
Ship	35	834
Appliance	18	537
Others (4 types)	26	518
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
Others (10 types)	56	1435
CREATIVEWORK	**200**	**4058**
Manga	36	391
TelevisionShow	33	902
VideoGame	32	948
Film	26	410
Single	25	436
Album	21	360
Anime	15	366
Others (3 types)	12	245
LOCATION	**200**	**4203**
Building	79	1978
Station	51	934
Museum	26	437
Library	9	159
School	8	169
Infrastructure	6	47
University	5	60
Others (6 types)	16	419
GROUP	**200**	**4459**
Company	168	3859
PoliticalParty	14	240
SoccerClub	12	275
Organisation	6	85
TOTAL	**800**	**16869**

(b) Japanese data

Table 5: 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**
FootballPlayer	13	448
SoccerPlayer	10	255
MartialArtist	9	212
BasketballPlayer	9	221
Politician	8	195
Person (Misc.)	6	243
Wrestler	3	42
Others (4 types)	7	134
CREATIVEWORK	**125**	**3892**
TelevisionShow	27	861
Film	22	547
Single	19	800
VideoGame	15	560
Album	14	453
Book	12	256
Comic	5	183
Others (6 types)	11	232
LOCATION	**2**	**100**
Stadium	1	50
Building	1	50
GROUP	**6**	**89**
PoliticalParty	3	14
Company	3	75
EVENT	**1**	**50**
WrestlingEvent	1	50
DEVICE	**1**	**50**
Appliance	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
Others (6 types)	7	122
CREATIVEWORK	**156**	**11310**
Single	39	3002
Album	33	2481
Film	28	1959
TelevisionShow	20	1655
Manga	15	817
VideoGame	7	463
Anime	5	339
Others (4 types)	9	594
GROUP	**6**	**510**
Company	6	510
TOTAL	**200**	**13430**

(b) Japanese data

Table 6: Statistics of **homographic** emerging entities and a burst of posts in the test data obtained from Twitter.

Name	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 7: Hyperparameters of our typing and context selection model. 'Context' means Context Network. 'Entity' means Entity Network. 'Meta' means Meta Network. 'CS' means Context Selection.