An Entity Disambiguation Approach Based on Wikipedia and Word Embeddings for Entity Linking in Microblogs

Tomoaki Urata, Akira Maeda

***Abstract*— The opportunity to read articles and microblogs on the Web to get information is more and more increasing. However, hyperlinks to entities do not often exist in such articles, and it is a troublesome task for the reader to look it up online. In this paper, in order to make it easy to look up entity information in microblog articles, we propose a method to extract entities in Japanese microblog, and to perform entity linking which links to entity information automatically. The method consists of three phases. First, we extract named entities, such as personal names, place names, organization names, etc. from a microblog article. Next, we disambiguate the extracted entities in order to make links to correct entity information. We use Wikipedia as the source of entity information to verify the usefulness of the proposed method. In our method, we extract Wikipedia articles related to ambiguous entities from microblog articles. Then, we extract related entities of the ambiguous entity using word2vec. We compare the text similarities of the Wikipedia articles of related entities and the Wikipedia articles of ambiguous entities. Finally, we get the correct Wikipedia article for each entity in the microblog article.**

***Index Terms*— Microblog, Named entity extraction, Word-sense Disambiguation, Wikipedia**

I. Introduction

The opportunity to read articles and blogs on the Web to get information is more and more increasing. However, hyperlinks to entities do not often exist in such articles, and it is a troublesome task for the reader to look it up online. In this paper, we propose an entity linking method to make it easy to look up information in microblog articles. Moreover, we solve the problem of entity disambiguation in order to make correct links to the entity information. we propose a method to extract entities in microblogs in Japanese, and to perform entity linking which links from entities to entity information automatically. Through this method, we can obtain information much more easily and conveniently. The image of the proposed method is shown in Fig. 1. In this figure, some entity names appear in a tweet. “スズキ (Suzuki)” means either a fish name or a company name. The proposed method automatically makes correct hyperlinks to the information of these entities even if the entity name has such ambiguities.

Entity Linking is the task of extracting descriptions which indicate entities in the text, and connect them with the corresponding item in external databases or knowledge bases. Entity linking is actively studied in the fields of information retrieval and natural language processing in recent years. Most of the existing research is targeted for newspaper articles and Web pages. Named entities such as personal names or organization names play an important role in understanding an article. Therefore, named entities are often regarded as important target in entity linking. Existing research on entity extraction in entity linking can be classified into two approaches, one is dictionary-based method, and the other is machine-learning-based method. We adopt the dictionary-based method.

We use a dictionary called NEologd[6], which is specialized in Japanese named entities. It is renewed at least twice a week. Therefore, we can extract named entities more flexibly than the method using machine learning.

For the knowledge base for linking information, we use Wikipedia. Wikipedia is an open content encyclopedia. Everyone can edit it freely and it is available in various languages. Wikipedia has contents in over 299 languages and the number of articles is over 46 million as of October 2017. It has not only huge number of articles but also high quality articles. Therefore, the reader can get information efficiently.

Fig. 1. The image of the proposed method

II. Related Work

A. Previous Study

We proposed the method of extracting entities in articles such as artists and to perform entity linking which links to artist Web sites to get artist information automatically[1]. The method consists of two phases. First, we extract artist names in music articles. An artist name is a named entity, and it is

Manuscript received December, 2017.

Tomoaki Urata is with a Graduate School of Information Science and Engineering, Ritsumeikan University, Kusatsu, Japan (e-mail: is0156ss@ed.ritsumei.ac.jp)

Akira Maeda is with the College of Information Science and Engineering, Ritsumeikan University, Kusatsu, Japan. (e-mail: amaeda@is.ritsumei.ac.jp).

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

necessary to distinguish artist names from other named entities. In order to achieve it, we prepared the training data of music articles in which artist names are manually tagged, and extract artist names using Support Vector Machine (SVM). Next, we choose a Wikipedia page of artist information for each extracted artist name. The method we proposed only uses Japanese music articles and extracted entities are only artist names. Besides, if there are ambiguous entities such as artists with identical names in music articles, the proposed method cannot handle it. Therefore, we propose a method that can get correct Wikipedia articles for any types of entities even if there are ambiguities. In addition, we propose an entity linking method for microblogs, in which the articles are very short and sometimes do not have enough context for disambiguation.

B. Entity Linking

Entity Linking is a task which extracts a description which indicates an entity in the text and connect it with an external database entry or an item in a knowledge base. Wikipedia is often used as the knowledge base for entity linking. The target entities to be linked are mainly personal names, organization names, company names, and place names.

Furukawa et al. [2] proposed the method of word sense disambiguation and linking from Japanese technical terms in the Japanese technical text to English Wikipedia articles. They extract technical terms in Japanese scientific literature by using a dictionary. The knowledge base is constructed by extracting the heading, the subheading, URL, body text and internal links from English Wikipedia articles and their sections. In our proposed method, we also use a dictionary when extracting entity names. However, we use a dictionary that is specialized in named entities.

Mihalcea et al. [3] introduced the use of Wikipedia as a resource for automatic keyword extraction and word sense disambiguation. They extract keywords by using the guideline of Wikipedia pages and calculate the importance of keywords. They used two approaches for word sense disambiguation. First approach is a knowledge-based approach, which relies exclusively on information drawn from the definitions provided by the sense inventory. Second approach is a data-driven method that integrates both local and topical features into a machine learning classifier. However, when there are new words or unknown words in the guideline, it is difficult to extract these words. Our method can extract these words by using a specialized dictionary.

Yamada et al. [4] proposed the method of enhancing the performance of the twitter named entity recognition task by using entity linking. Their method is based on machine learning and uses the knowledge obtained from several open knowledge bases. Also, they proposed the method of word-sense disambiguation which matches an ambiguous entity in twitter and a corresponding Wikipedia article title, the page titles of the Wikipedia article that redirects to the page of the entity, and anchor texts in Wikipedia articles. Our method uses a dictionary when extracting entities. We propose the method of entity disambiguation by using surrounding unambiguous entities. We use the entities that appear at the nearest from the target ambiguous entity.

C. Named Entity Extraction

Named entity extraction is the method which extracts named entities from text. It extracts proper nouns such as place names, personal names, organization names, and numerical representations such as date, time, and currency. Information Retrieval and Extraction (IREX) defined a set of named entity categories in their workshop. They defined 8 categories of named entities. They are organization name, person name, location name, date, time, money, percent, and artifact name. There are two methods for extracting named entities. First, a method based on rules created manually. Second, a method that uses machine learning. The methods based on Support Vector Machine (SVM), maximum entropy and Conditional Random Field (CRF) are proposed as the method of machine learning. In the methods of extracting named entities by machine learning, usually they divide the input text into analytical units and then connecting one or several tokens that constitute a named entity.

Yamada et al. [5] proposed a named entity extraction method based on grouping of tokens using SVM. They used a word as the unit of analysis. They used the word itself, part of speech, and the type of the character as features. They used CRL (Communications Research Laboratory) named entity data in their experiment. The data contains 1,174 Mainichi Newspaper articles and about 11,000 sentences have named entities tagged. They achieved F-measure of about 83% in their experiment. This result shows that SVM is effective for the grouping of tokens.

Sato [6] proposed the method of updating a named entity dictionary continuously to correspond to new words and unknown words which existing methods cannot correspond. The dictionary is called “NEologd”. We propose the method using this dictionary for entity linking in twitter articles in order to extract the entities that existing methods cannot extract.

III. Proposed Method 1

We propose two methods of entity disambiguation in this paper. In our proposed method 1, first we extract entity names from a twitter article. Next, we get the correct links to entity information using a method of entity disambiguation. Our method consists of three steps. First, we extract entity names from a twitter article using a Japanese morphological analyzer MeCab[7] with the named entity dictionary NEologd. Second, we solve the problem of ambiguous entities by using Word2vec[10] from the target entity and the nearest entities in the article. Finally, we get the correct entity information from Wikipedia. We show the overview of our proposed method in Fig. 2.

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

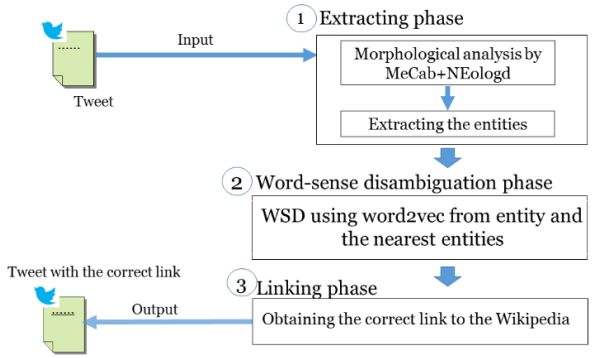

Fig. 2. Overview of our proposed method

A. *Extraction Phase*

New words or unknown words often appear in microblog such as twitter. Also, such a word is often an entity which we want to extract. Therefore, we extract such entities by using MeCab with the dictionary NEologd. All of the nouns in the tweet are matched with Wikipedia article names, and if the word has the disambiguation page, then the word is assumed as an ambiguous entity.

B. *Entity Disambiguation Phase*

We take five steps to perform entity disambiguation. We show the overview of the entity disambiguation phase in Fig.3.

1) Obtaining the Candidate Entities

In Wikipedia, there are disambiguation pages for ambiguous entities. We show an example of the disambiguation page for an ambiguous entity in Wikipedia in Fig.4. In this tweet, the Japanese word “スズキ (Suzuki)” is an ambiguous entity. When we search this word as a query in Wikipedia, the candidate entities which correspond to “スズキ” can be obtained. In this case, “スズキ (Suzuki)” is either a fish name or a motorcycle and automobile maker in Japan. Thus, we obtain the Wikipedia articles of the candidate entities from Wikipedia disambiguation page, which will be used to disambiguate the entities in the next step.

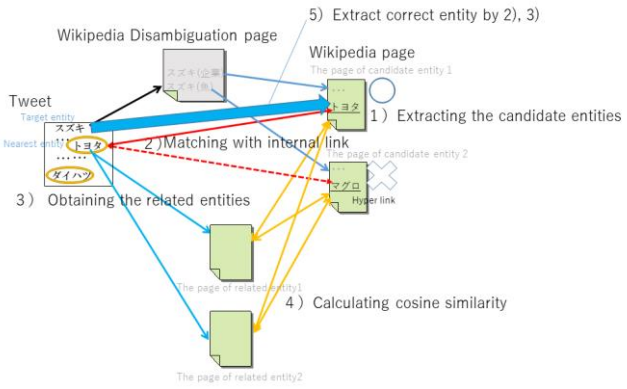

Fig. 3. Overview of the word-sence disambiguation phase

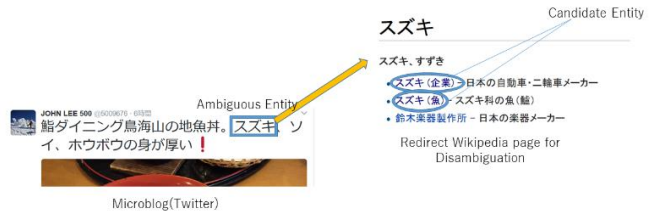

Fig. 4. Example of disambiguation page for an ambiguous entity in Wikipedia

2) Matching with internal links in Wikipedia

We compare all of entities in the tweet and each Wikipedia article of the candidate entity and match these entities with internal links in the Wikipedia articles of the candidate entities. If it matches at least one internal link, we give the score “1”. If it does not match, we give the score “0”.

3) Obtaining the related entities

We define the nearest entity as the nearest unambiguous nouns from the target entity that do not have the disambiguation page in Wikipedia. We get the related entities of the nearest entities using word2vec. We use all Japanese Wikipedia articles to train word2vec. We get the top five related entities for each nearest entity and those Wikipedia articles. We show the example of nearest entities and the target entity in fig.5. In fig.5, three target Japanese entity “スズキ(Japanese sea perch)” appear in the tweet. The nearest entity in the first target entity is “一本 (a piece)”, the second one is “塩レモン (salted lemon)”, and the third one is “明日 (tomorrow)” and “料理 (meal/dish)”.

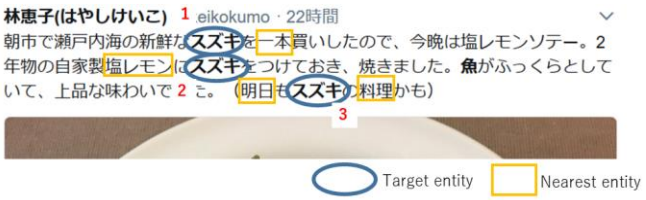

Fig. 5. Example of nearest entities and target entity

4) Calculating Similarity

We calculate cosine similarities between the Wikipedia pages of related entities and the Wikipedia pages of the candidate entities. Equation (1) is the formula of the cosine similarity used in our method. The vectors A and B are the term frequency vectors of nouns in the candidate entities Wikipedia page and related entities Wikipedia page, respectively.

$$\text{similarity}(A,B) = \frac{A \cdot B}{\|A\|^2\|B\|^2} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \quad (1)$$

5) Determining Correct Entity by 2),4)

We multiply the results of step 2 and 4 above and we select the entity with the highest score as the correct Wikipedia article for the entity.

IV. Proposed Method 2

In our proposed method 2, extracting phase and linking phase are same to method 1, however word-sense disambiguation phase is different. We show the overview of

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

the entity disambiguation phase of the proposed method 2 in Fig.6. We take seven steps to perform entity disambiguation in the proposed method 2.

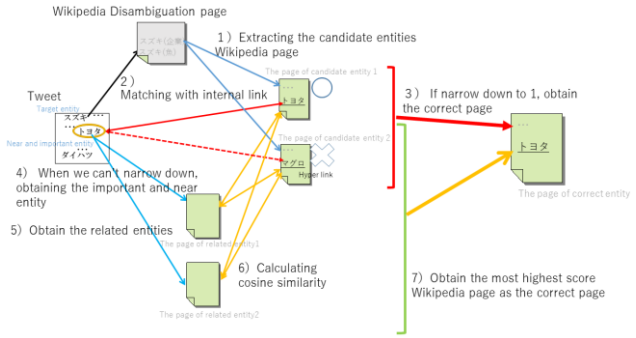

Fig. 6. Overview of the word-sence disambiguation phase of proposed method 2

1) Obtaining the Candidate Entities

This step is the same with Ⅲ-B-1).

2) Matching with internal links in Wikipedia

Same as III-B-2), we compare all of entities in the tweet and each Wikipedia article of the candidate entity and match these entities with internal links in the Wikipedia articles of the candidate entities.

3) Extracting correct entity through internal links

When we narrow down the candidate entities to one article through step 2), which means that it matches just one internal link, we decide this article as a correct Wikipedia page and the method finishes. When we cannot narrow down to one article in step 2), we go to step 4).

4) Obtaining the important and nearby entity

We calculate the importance of words by using a lot of tweets and OkapiBM25+[8]. OkapiBM25+ is an extension of OkapiBM25[9]. OkapiBM25 is a bag-of-words retrieval function that ranks a set of documents based on the query terms appearing in each document, regardless of the inter-relationship between the query terms within a document. OkapiBM25+ was developed to address one deficiency of the standard BM25 in which the component of term frequency normalization by document length is not properly lower-bounded; as a result of this deficiency, long documents which do match the query term can often be scored unfairly by BM25 as having a similar relevancy to shorter documents that do not contain the query term at all. Equation (2) is the formula of the OkapiBM25+.

$$\text{score}(D,Q) = IDF(q_i)\left(\frac{TF(q_i,D)\cdot(k_1+1)}{TF(q_i,D)+k_1\cdot\left(1-b+b\cdot\frac{|D|}{avgdl}\right)}+\delta\right) \quad (2)$$

D is a Document. $TF(q_i, D)$ is q_i's term frequency in the document D, $|D|$ is the length of the document D in words, and *avgdl* is the average document length in the text collection from which documents are drawn. k_1 and b are free parameters, usually chosen as $k_1 \in [1.2,2.0]$ and b = 0.75. δ is 1.0. $IDF(q_i)$ is the inverse document frequency (IDF) weight of the query term q_i. It is computed as Equation (3).

$$IDF(q_i) = log\frac{N}{d_w} \quad (3)$$

N is the total number of documents in the collection. and d_w is the number of documents containing q_i.

After calculating the importance of words, we give the weight to the importance based on the distance between the target entity and nearby entities. The weights formula is shown in Equation (4).

$$weight(q_i) = 1 - k\cdot|l| \quad (4)$$

k is the free parameter. ||l| is the distance between the target entity which has ambiguity and q_i.

Finally, we get the important and nearby words by calculating the score using Equation (5).

$$word = \max(score(D,q_i)\cdot weight(q_i)) \quad (5)$$

5) Obtaining the related entities

We get the related entities of the important and nearby entity using word2vec. We get the top five related entities for each important and nearby entity and those Wikipedia articles.

6) Calculating Similarity

We calculate cosine similarities between the Wikipedia pages of related entities and the Wikipedia pages of the candidate entities.

7) Determining Correct Entity

We select the entity with the highest score as the correct Wikipedia article for the entity.

V. Experiments And Discussion

In this section, we describe the experiments to verify the usefulness of the proposed methods and discuss the results. We experiment 4 methods to compare the precision with proposed method. First method is Ⅲ-B-2). If they math, we extract the matched Wikipedia page as a correct entity article. Second method is calculating cosine similarity by Ⅲ-B-4). The third method is proposed method. The fourth method is proposed method2. We collect 50 tweets which has word sense disambiguation. When we calculate OkapiBM25+, we collect 1,046,967 tweets and the average document length is 21.3. k_1 is 1.5. When we weigh the score, k is 0.05.

A. Tool for Extracting Related Entities

1) Word2vec[1]

We use "word2vec" as a tool of extracting related entities. The word2vec tool takes a text corpus as input and produces the word vectors as output. It first constructs a vocabulary from the training text data and then learns vector representation of words. For example, if you enter the word "france", it will display the most similar words and their distances to "france". We use all Japanese Wikipedia article as a text corpus. Mikolov et al. [10] propose model architectures for computing continuous vector representations of words from very large data sets. They show state-of-the-art performance on their test set for measuring syntactic and semantic word similarities.

[1] https://code.google.com/archive/p/word2vec/

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

B. Experiments of Entity disambiguation

We conducted the experiments using the 3 kinds of WSD entity. They are company name, person name, animal name. We conducted the experiments using 5 pieces of WSD words. Local name is "アフリカ(Africa)". Person names are "田中理恵(Rie Tanaka)" and "ジョン・ウィリアムズ(John Williams)". Animal names are "ライオン(lion)" and "スズキ(Suzuki)".

1) Test in company name

We conducted the experiment the Japanese word "アフリカ" as an ambiguous company name. "アフリカ" have 5 meanings in Japanese. The result is shown in TABLE I. All of 10 tweets contain the mean of Country Name.

TABLE I
THE RESULT OF "アフリカ"

	Method1	Method2	Proposed Method1	Proposed Method2	Match Word
Tweet1 (Country)	○	○	○	○	ナイジェリア (Nigeria)
Tweet2 (Country)	○	○	○	○	アパルトヘイト (Apartheid)
Tweet3 (Country)	×	○	×	○	
Tweet4 (Country)	×	○	×	×	
Tweet5 (Country)	×	○	×	○	
Tweet6 (Country)	×	○	×	○	
Tweet7 (Country)	×	○	×	○	
Twwet8 (Country)	×	○	×	○	
Tweet9 (Country)	×	○	×	○	
Tweet10 (Country)	×	○	×	○	
Accuracy	0.2	1.0	0.2	0.9	

2) Test in person name

We conducted the experiment the person names "田中理恵(Rie Tanaka)" and "ジョン・ウィリアムズ(John Williams)" as ambiguous words.

a) Test in "田中理恵"

"田中理恵" has two meanings in Japanese. They are a voice actor name or a gymnast. We collect 9 tweets which contains voice actor's names and 1 tweet which contain gymnast's name. The result is shown in TABLE Ⅱ.

TABLE Ⅱ
THE RESULT OF "田中理恵"

	Method1	Method2	Proposed Method1	Proposed Method2	Match Word
Tweet1 (Voice actor)	○	○	○	○	声優 (Voice actor)
Tweet2 (Voice actor)	×	○	×	○	
Tweet3 (Voice actor)	×	○	×	×	
Tweet4 (Gymnast)	×	○	×	○	
Tweet5 (Voice actor)	×	○	×	○	
Tweet6 (Voice actor)	○	○	○	○	アニメ (Animation)
Tweet7 (Voice actor)	×	○	×	○	
Twwet8 (Voice actor)	×	×	×	×	
Tweet9 (Voice actor)	○	○	○	○	声優 (Voice actor)
Tweet10 (Voice actor)	×	○	×	×	
Accuracy	0.3	0.9	0.3	0.7	

b) Test in "ジョン・ウィリアムズ"

"ジョン・ウィリアムズ" has five meanings in Japanese. They are a music creator, a guitarist name, actor name, political scientist name, and missionary name. We collect 6 tweets which contains music creator name and 2 tweets which contains guitarist name and 2 tweets which contains actor name. The result is shown in TABLE *Ⅲ*.

3) Test in animal name

We conducted the experiment the animal names "ライオン(lion)" and "スズキ(Suzuki)" as ambiguous words.

a) Test in "ライオン"

"ライオン" has seven meanings in Japanese. They are a n animation song name, a band name, Japanese company name, a battle ship name, a heraldry name, an animal name, and a Japanese artist's song name. We collect 8 tweets which contains animal name and 1 tweet which contains company name and 1 tweet which contains animation song name. The result is shown in TABLE *Ⅳ*.

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE Ⅲ
THE RESULT OF "ジョン・ウィリアムズ"

	Method1	Method2	Proposed Method1	Proposed Method2	Match Word
Tweet1 (music creator)	○	○	○	○	映画音楽 (film music)
Tweet2 (music creator)	○	○	○	○	映画音楽 (film music)
Tweet3 (music creator)	×	×	×	○	
Tweet4 (music creator)	○	○	○	○	映画音楽 (film music)
Tweet5 (music creator)	○	×	○	○	映画音楽 (film music)
Tweet6 (music creator)	×	○	×	○	
Tweet7 (guitarist)	×	○	×	○	
Tweet8 (guitarist)	×	○	×	○	
Tweet9 (actor)	○	×	○	○	俳優 (actor)
Tweet10 (actor)	○	×	○	○	俳優 (actor)
Accuracy	0.6	0.6	0.6	1.0	

TABLE Ⅳ
THE RESULT OF "ライオン"

	Method1	Method2	Proposed Method1	Proposed Method2	Match Word
Tweet1 (animal)	×	○	×	○	
Tweet2 (animal)	○	○	○	○	亜種 (subspecies)
Tweet3 (animal)	○	○	○	○	ケニア (Kenya)
Tweet4 (animal)	×	×	×	×	
Tweet5 (animal)	×	○	×	○	
Tweet6 (animal)	×	○	×	○	
Tweet7 (animal)	×	○	×	×	
Twwet8 (animal)	○	○	○	○	トラ (tiger)
Tweet9 (animation song)	○	×	○	○	マクロスF (MACROSS Frontier)
Tweet10 (company)	×	×	×	○	
Accuracy	0.4	0.7	0.4	0.8	

b) Test in "スズキ"

"スズキ" has two meanings in Japanese. They are a fish name and company name. We collect 4 tweets which contains fish name and 6 tweets which contains company name. The result is shown in TABLE Ⅴ.

TABLE Ⅴ
THE RESULT OF "スズキ"

	Method1	Method2	Proposed Method1	Proposed Method2	Match Word
Tweet1 (fish)	×	○	×	○	
Tweet2 (company)	×	×	×	○	
Tweet3 (company)	×	×	×	×	
Tweet4 (company)	×	×	×	○	
Tweet5 (company)	×	○	×	×	
Tweet6 (fish)	○	○	○	○	釣り (fishing)
Tweet7 (company)	×	○	×	○	
Twwet8 (company)	×	×	×	○	
Tweet9 (fish)	×	○	×	○	
Tweet10 (fish)	×	○	×	○	
Accuracy	0.1	0.6	0.1	0.8	

C. Discussion

We experimented with 50 tweets. The accuracy of method 1 was 32%, the method 2 was 76%, the proposed method was 32%, the proposed method 2 was 84%. The proposed method 2 was the most accurate result in the experiment. First we discus the method 1. It is assumed that low accuracy of the method 1 is caused by two reasons.

First, when we match an entity in the tweet and the hyperlinks in candidate entities' Wikipedia page, we do not target the bold face entities in the Wikipedia page. Those entities are often important entities in Wikipedia. Therefore, if we target the bold face entities in the Wikipedia page, we could improve the accuracy of method 1.

Second, when we match the entity in the tweet and the hyperlinks in candidate entities' Wikipedia page, we do not target the entity which is composed of several morphemes. When we extract entities using NEologd, the entity is extracted as one entity. However, when we extract entities without using NEologd, entities are divited into several morphemes by the morphological analyzer. Some of these entities are important for matching.

Next, we discus about the method 2. The tweet which did not extract correct Wikipedia page had the nearest entity which is not related to the target entity. However, the entity which is far from the target entity is sometimes an important

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

entity for word sense disambiguation. Therefore, method 1 could extract correct Wikipedia page, but method 2 could not extract correct entity.

Finally, we discuss the proposed method 2. We were able to obtain the important entities which are near from the target entity. However, when the important entity is located far from the target entity, we could not obtain that entity as an important entity. Besides, when the target entity appears more than twice in a tweet, we could not obtain the important entity correctly.

D. Future Work

In the proposed method, we extract the correct Wikipedia page using nearby and important entities from the target entity. However, we could not obtain important entities when they are located far from the target entity. Therefore, if we set the special process to the entity which has maximum score, we might be able to obtain these important entities. In addition, it is important for the proposed method 2 whether matching entity in the tweet and the hyperlink in the candidate entities' Wikipedia page exists or not. Therefore, when we target the bold face entities in the Wikipedia page, we could improve the accuracy.

VI. Conclusion

In this paper, we proposed a method for extracting entity names from microblog and disambiguating these entities using entities near from that target entity and their related entities obtained using word2vec. In the experiments, we were able to get the correct entity accurately. For more improvement, we are planning to target the bold face entities in the Wikipedia page when we match entity in the tweet and the hyperlinks in candidate entities' Wikipedia page.

References

[1] Tomoaki Urata, Akira Maeda, "Entity Linking of Artists Names in Japanese Music Articles", In Proceedings of the 7th International Conference on E-Service and Knowledge Management, 2017

[2] Tatsuya Furukawa, Takeshi Sagara and Akiko Aizawa,"Semantic Disambiguation for Cross-Lingual Entity Linking". Journal of Japan Society of Information and Knowledge, vol. 24 No. 2, pp. 172-177, 2014, (in Japanese)

[3] Rada Mihalcea, Andras Csomai, "Wikify! Linking Documents to Encyclopedic Knowledge", Conference on information and knowledge management, pp. 233-242, 2007

[4] Ikuya Yamada, Hideaki Takeda, Yoshiyasu Takefuji "Enhancing Named Entity Recognition in Twitter Messages Using Entity Linking" In Proceedings of the ACL 2015 Workshop on Noisy User-generated, pp. 136–140, 2015

[5] Taku Kudo and Yuji Matsumoto, "Chunking with Support Vector Machines". In Proceedings of The Second Meeting of the North American Chapter of the Association for Computational Linguistics for Computational Linguistics on Language technologies (NAACL2001), pp. 1-8, 2001

[6] Toshinori Sato "mecab-ipadic-NEologd: Neologism dictionary for MeCab"<https://github.com/neologd/mecab-ipadic-neologd> (accessed 25 December 2017)

[7] Taku Kudo, Kaoru Yamamoto, Yuji Matsumoto "Applying Conditional Random Fields to Japanese Morphological Analysis", Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004), pp.230-237,2004.

[8] Yuanhua Lv, ChengXiang Zhai, "Lower-Bounding Term Frequency Normalization", International Conference on Information and Knowledge Management (CIKM), pp 7-16,2011.

[9] S.E.Robertson, S.Walker,S.Jones, M.M.Hancock-Beaulieu, M.Gatford. "Okapi at TREC-3", Proc. of TREC-3, pp.109-126, 1995

[10] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, "Efficient Estimation of Word Representations in Vector Space" In Proceedings of the International Conference on Learning Representations (ICLR), 2013.

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)