*International Journal of Engineering & Technology*, 7 (4.38) (2018) 11-16

SPC

**International Journal of Engineering & Technology**

*Website: www.sciencepubco.com/index.php/IJET*

***Research paper***

# A Hybrid Bootstrapping Approach for developing Odiya Named Entity Corpora from Wikipedia

**Sitanath Biswas[1], Sujata Dash[2]**

*North Orissa University*
**Corresponding author E-mail: sitanathbiswas2006@gmail.com*

**Abstract**

Named Entity Recognition (NER) is considered as very influential undertaking in natural language processing appropriate to Question Answering system, Machine Translation (MT), Information extraction (IE), Information Retrieval (IR) etc. Basically NER is to identify and classify different types of proper nouns present inside given file like location name, person name, number, organization name, time etc. Although huge amount of progress is made for different Indian languages, NER is still a big problem for Odiya Language. Odiya is also a resource constrained language and till today, this is very tough to find out a large and accurate corpus for training and test. Therefore in this paper, we have utilized Wikipedia to develop a huge Odiya corpus of annotated name entities which is quite efficient to be training dataset further. After evaluation, we have got a very promising result with a F-score of 78.89.

***Keywords***: *Named Entity Recognition, NER, Wikipedia, Machine Translation, Information Extraction, Information Retrieval.*

## 1. Introduction

Named Entity Recognition (NER) is considered as a very important job under natural language processing appropriate to Question Answering system, Machine Translation (MT), Information extraction (IE), Information Retrieval (IR) etc. Basically NER is to identify and classify each and every appropriate nouns present in a text as location name, person name, company name, digit, time etc [1]. For last twenty five years, NER is a dynamic field of research in the field of NLP. But NER still remains a big problem for Odiya Language. Odiya is also a resource constrained language and till today, this is tough to search for a large and accurate corpus for training and test. The greatest challenge to develop Multilingual NER system for Indian Languages is:

Morphologically rich –As Odiya language is morphologically very rich, it is very difficult to identify the root word, and therefore it requires morphological analysers [1].

Capitalization feature - In English, capitalization plays a big role to identify NEs but it is not found in Odiya languages [4].

Ambiguity – thousands of ambiguities are present in common and proper nouns.

Spell variations – When it comes to web, then a same thing can be spelled differently in different domain.

From last two decades, NER is the prime attention of NLP researchers [26, 27]. The initiative in the field of NER was taken during Message Understanding Conferences (MUCs) [26, 27], during the development of GATE system. Precise finding of NEs was reported and later standardized by the inventers [3]. NER was also got tremendous importance during the development of Information Extraction System [29], question-answering systems [30], machine translation [31]. In the early times, researchers were using finite state automata to match against a series of words common regular expression comparer. LaSIE-II by Sheffield University [32], NetOwl by ISOQuest [33] and LTG by University Of Edinburgh [34] are the English NER. These systems were actually based on rule and therefore these systems are not robust and have issue like portability. Developing rule based system is quite expensive because every time we use new text as input, we need to manipulate the existing rule to manage the optimal performance. In recent days, machine-learning (ML) [21] approaches are extensively implemented in NER. The basic advantage of using ML is that one can train it easily, ML is very adapting in nature to various domains and languages and to maintain is very less expensive [20]. Different machine learning techniques [20] used in NER so far is Hidden Markov Model (HMM) [36], Maximum Entropy (ME) [20] model, University of New York's MENE [38], in the New York University's system, Decision Tree was implemented [39], CRF [40]. Shallow parsing approach by Pattern-directed for NER in Bengali was reported by Bandyopadhyay and Ekbal [16]. The paper describes two different model of NER, where one model uses lexical contextual designs and additional uses language based characteristics with lexical contextual patterns of same set. A NER system using HMM was reported by Ekbal [16], to handle unknown entities, the author uses maximum number of contextual information and named entity suffixes during probability (emission). More recent contributions in Bengali NER can effortlessly be get in [4] [11] accompanied by CRF, and the SVM technique, commonly. Those NER tools were developed by using various contextual features and orthographical word based characteristics in association with a variety of characteristic took out, out of the gazetteers. The NER work on the language Hindi was reported by Mc-Callum and Li [6] using CRF technique which implements a process known as characteristic ordination to build the characteristics automatically to grow the conditional likelihood. In various papers, authors have used Wikipedia for extraction of information and Named Entity Recognition. Yago [22] and DB-pedia [3] extracted useful details from the organized portions like info-boxes, lists, groups and other things). Suchanek [22] had developed a concept of hierarchical relation by depicting a derived object to WordNet [12]. Ruizcasado and others [17] has introduced a technique for

automated deriving and generalizing of extracted design for connotation associations (meronymy/ holonymy, hyponymy/ hyperonymy,) through ordinary Wikipedia in English, implementing and expanding WordNet. The derived sequences are generalized by using an algorithm which uses lowest edit space, implementing a depiction similar to the algorithm introduced. Culotta and other authors [8] introduced a prototype to implement data mining (DM) and information extraction (IE), shown on various Wikipedia essays. They had applied CRFs, implementing both context and relation characteristics. Nguyen and other members [14] exhibit relational removal through Wikipedia essay complementary texts implementing tree mining (dependency) and machine learning algorithm (supervised) using SVM classification. The authors have utilized a traditional co-referential resolution algorithm utilizing exceptional features of Wikipedia article. They had also implemented traditional named entity type recognition depending upon supervised algorithm for classification of Wikipedia articles representing to entities inside the relations. Suchanek and other authors [21] represented PORE, the algorithm for circumstances integrating only unlabeled and positive instances, pertained to semi-automatic Information Extraction from free text in Wikipedia essays. This technique basically implements the Support Vector Machine classification technique, and implements bootstrap approach, powerful negative recognition and trans-ductive deduction.

In this paper we have used Wikipedia, an online, freely accessible, accurate, fast and ever growing free resources to develop a named entity annotated corpus which may be further utilize like data set training towards any NER system. For each and every article in Wikipedia has a link and every link corresponds to a named entity and that named entity which considered as actually the link, takes you to another related and appropriate link or named entity. In this way, we can identify and extract millions of sentences out of the Wikipedia and create an enormous corpus for Odiya language that may be later utilize like a training data for any standard named entity recognition system. For evaluation purpose, we have taken standard Odiya corpora and our corpus which is created through Wikipedia and tested. Therefore in this manner, we can develop many general purpose or domain specific corpora very easily and effectively without the help of manual annotation.
In last 25 years tremendous research have been taken place in the field of NER, unfortunately for Odiya language, the research is active from last 10 years only. Considering the quality research by other scientists round the globe, initially Odiya NER was developed by using handcrafted rules, Gazetteers, Machine learning Techniques etc. But the major bottleneck was lack of large and quality corpus. Although Machine Learning was showing very promising result [16] for language like English, Portuguese, Hindi, Tamil [4], but Odiya language was not getting good result because of the scarsity of huge corpus.

The remaining part of our paper is arranged like this. Section number 2 presents creation of Named Entity corpora from Wikipedia. Section number 3 describes the classification of Wikipedia articles. Section number 4 gives sentence selection and extraction, Section number 5 describes the evaluation of experiment and section number 6 gives conclusion and future work.

## 2. NE corpora from Wikipedia

Wikipedia is an online repository or Encyclopedia which is written by billions of its users, and which includes more than 5.2 million articles in English and other various languages. Wikipedia is also freely accessible, accurate, fast and ever growing free resources to develop a named entity annotated corpus which may be further utilize like data set training for any NER system [7]. For each and every article in Wikipedia has a link and every link corresponds to a named entity and that named entity which considered as actually the link, takes you to another related and appropriate link or named entity. In this way, we can identify and extract millions of sentences out of the Wikipedia and create an enormous corpus for Odiya language that may be later utilize like a training data for any standard named entity recognition system. Since more than 81% of Wikipedia articles describe the different topics which fall under various traditional entity classes. Moreover various link of Wikipedia represent in gold-standard entity annotations in NER training corpora [9]. The major advantage of using Wikipedia is that it supports a crucial concept called word sense disambiguation. The link which corresponds to NEs, it also disambiguates the referent. For example “Harishankar” a person name, from “Harishankar” as place name, “Ganga” is a river; from “Ganga” is a name of a lady. Following are the steps by which we derive the named entity corpus from Wikipedia:

To create entity classes, classify all Odiya articles first.

- Split all Odiya Wikipedia articles into independent sentences.
- As per the link target, label all Odiya named entities.
- Selected sentences will be included in the corpus.

This approach is not at all language or domain specific. Therefore it can be implemented to any language and any domain. For evaluation purpose, we have used CoNLL standard i.e. PER, LOC, ORG and MISC [2].

## 3. Wikipedia Article Classification:

First of all, Wikipedia article must be classified to a fixed set of entity categories so that the labeling of links as per to their target. In our work we have used a hybrid bootstrapping technique towards classifying purpose. We have considered two cases here, one is unknown category and one is known category of NEs on the basis of the heuristic knowledge. During bootstrapping, mapping is learnt and entity classes are assigned. The major challenge here is to find out the non-entity articles which are actually diverse and large in Wikipedia. That is why we first try to classify all articles as non-entity. For general classification purpose, we use bootstrapped heuristics to extract feature from articles.

In order to find out the category noun, we have used a standard Odiya POS Tagger. Basically POS Tagger helps for tagging and chunking. Another heuristic feature we have used here is called suffix stripping approach. Suffix stripping approach was successfully implemented for English by Paice/Husk [4] any many other authors. Similarly for Indian languages, suffix stripping algorithm was implemented by [5], [46]. Ramanathan, Rao [46] and Larkey [47], has implemented suffix stripping approach with predefined 27 suffixes for number, gender etc successfully. Suffix stripping was also used in morphological analyzer for Bengali language by Dasgupta and Nag [47].

The advantage of suffix stripping algorithm is that it never depends on the stored database or look-up table. It generally works on specially designed hand crafted rules. The algorithm uses these rules as a route to find out the root word. \for example, there is a Named entity called “Rama*ku”*, the algorithm remove the suffix “*ku*” and extract the root word “Rama”. Following is the suffix stripping algorithm:

*Start*
*Step 1: Input the token received from Wikipedia*
*Step 2: Find out the suffix in the token*
*Step 3: If the token has the suffix then*
*Step 4: Eliminate the suffix and extract the root word*

*Step 5: Display the result*
*Stop*

Algorithm 1: Suffix stripping approach

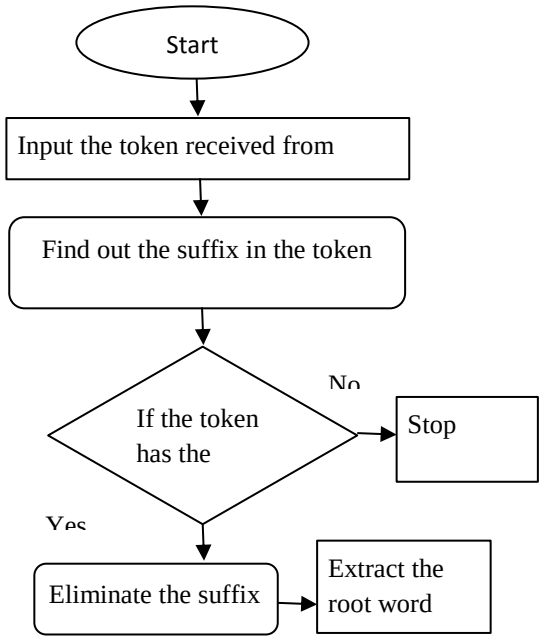


**Fig.1:** Flowchart for Suffix stripping approach

**Fig. 2:** Wikipedia Article. Red oval shape indicates NE, represented by link.

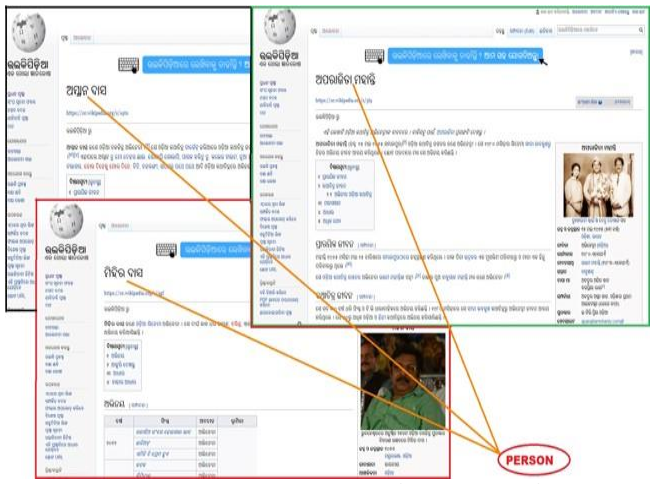


**Fig.3:** Named entity derived and tagged from the text in figure 1.

We have identified potential Odiya suffix information which plays a crucial role to find out the definition nouns. The partial list Odiya suffix is given below:

**Table 1:** Partial list of Odiya suffix

| Suffix | Suffix in Odiya |
|---|---|
| re | ରେ |
| ru | ରୁ |
| ku | କୁ |
| pai | ପାଇଁ |
| bina | ବିନା |
| tharu | ଠାରୁ |
| sahita | ସହିତ |
| e | ଏ |
| ila | ଇଲା |
| iba | ଇବା |
| anta | ଅଂଟା |
| uchi | ଉଚି |
| uthila | ଉଠିଲା |
| uthiba | ଉଠିବା |
| uthanta | ଉଥାନ୍ତା |
| ichi | ଇଛି |
| ithila | ଇଥିଲା |
| ithiba | ଇଥିବା |
| ithanta | ଇଥାନ୍ତା |

### 3.1.Bootstrapping Approach:

The primary benefit of bootstrapping approach is that it avoids huge numbers of hand crafted rules and bag of words method which is practically difficult to develop and manage. For the purpose of general classification, features were accumulated from Wikipedia articles and each feature was mapped to the equivalent entity class. We have considered two types of noun here, category noun and definition noun. As per the diagram, we have used bootstrapping method as our primary classification process. In order to map the entity classes with category noun and definition noun, we have used hand labelled data. Here we have used a feedback link to use the optimal result of one classification to create heuristic mapping for the next. As maximum article belongs to multiple categories, therefore the iteration of bootstrap produces more optimistic classification of Wikipedia contents.

The mappings can be inferred like following: If a set of artifact and their category are given, the no. can be count of the number of occurrences of every attribute with respect to the category. Like every candidate noun N (may be unigram or bigram), with the category k that is very frequently linked is confirmed. If n number of categorized texts backing the align N→k and m texts controvert this, therefore we secure the aligning

if $n \geq t$ and $\frac{m}{n+m} < p$, towards few fixed thresholds 't' and 'p'.

We had considered p = 0.25, towards cost for't'.

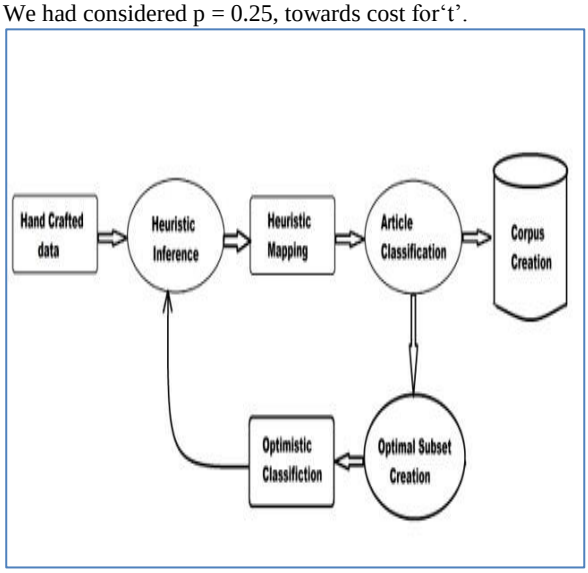


**Fig. 4:** Article classification by bootstrapping approach

## 4. Sentence selection and extraction:

In order to prepare NER training corpus from Wikipedia article, first we needs cleaning of noise, sentence separation and tokenization [8] as Wikipedia is not created by simple text format, rather it is created by a special marked up language. Before tokenization, we have used standard algorithm (unsupervised) for identifying the boundary of the sentence [8]. The source code for standard unsupervised algorithm for sentence detection algorithm is freely available at https://www.nltk.org/_modules/nltk/tokenize/punkt.html. The algorithm was developed by Tibor Kiss and Jan Strunk. The source code is language independent in nature. After for sentence boundary detection, we had tokenize the text by using the code which is freely available and Unicode compatible at https://www.nltk.org/_modules/nltk/tokenize/regexp.html.

For identifying dates, months, year, days of the week, we have used simple regular expressions. To identify the tittles, we have created a list of the Odiya titles which is described in the below mentioned table. Another approach is also used to identify surnames. If a certain link becomes visible right before the link towards a target "person", we infer that this must be the surname and can be considered for inclusion in our corpora leaving a Named entity tag.

**Table 2:** Partial list of Odiya title

| Title | Title in Odiya |
|---|---|
| Sri | ଶ୍ରୀ |
| Sriman | ଶ୍ରୀମାନ |
| Srimati | ଶ୍ରୀମତୀ |
| Mananiyo | ମାନନୀୟ |
| Mananiya | ମାନନୀୟ |
| Srijukta | ଶ୍ରୀଯୁକ୍ତ |
| Mahashaya | ମହାଶୟ |
| Shikhyak | ଶିକ୍ଷକ |
| Mukhyamantri | ମୁଖ୍ୟମନ୍ତ୍ରୀ |
| Mantri | ମନ୍ତ୍ରୀ |
| Guru | ଗୁରୁ |
| Doctor | ଡ଼କ୍ଟର |
| Sikhika | ଶିକ୍ଷିକା |

## 5. Evaluation and Discussion:

The accomplishment of our methods was studied with regard to standard Recall, Precision and *F*-measure:

*Precision = (valid positives) / (valid positives + invalid positives)*
*Recall= (valid positives) / (valid positives + invalid negatives)*
*F measure= (2* Precision * Recall) / (Precision+ Recall) [14]*
Where:

• *Valid* positives indicates the quantity of Named Entities classified correctly
• *Invalid* positives represent the quantity of Named Entities classified for non NEs
• *Invalid* negatives represent the quantity of Named Entities not categorized for correct Name Entities

We have taken three different models for evaluation purpose: training accompanied by Wikipedia data, training accompanied by hand crafted annotated data and training accompanied by both associated data. We have used Conditional Random Field (CRF) tagger [13], which is customizable and freely available at http://crfpp.sourceforge.net. For better result, we had utilized a standard Odiya gazetteer, contextual information and orthographic features [1]. The findings are given in terms of precision, recall and F-measure in table number 4. As Wikipedia data is freely available for download, our experiment was performed on 62,000 sentences derived from Wikipedia although 19 millions sentences are available in Wikipedia. We could not extract more Wikipedia sentences due to time and memory requirement. Although with little modification to the existing model, it is very much possible to extract more sentences. In this work, for standard Odiya corpus,

we have used IJCNLP 2008 Shared Task data on (NERSSEAL) South and South East Asian Languages and also manual annotated data for Odiya.

**Table 3:** Corpora used for evaluation

| Corpus | No. of training data | No. of test data |
|---|---|---|
| Standard Odiya Corpus | 52550 | 45130 |

**Table 4:** Comparison of Evaluation.

| Training Corpus | Precision | Recall | F-Measure |
|---|---|---|---|
| Standard Odiya Corpus | 69.25 | 78.13 | 72.39 |
| Wikipedia Corpus | 77.29 | 83.15 | 78.89 |

As the table 4 indicates, the result for Wikipedia corpus is outperforming over standard Odiya corpus. Where standard Odiya corpus is giving 72.39% F-measure, the F-measure for our Wikipedia corpus is very much promising i.e. 78.89%. The precision, recall and overall F-measure can be enhanced if we could extract more and more sentences from Wikipedia which requires more time and memory. We have derived only 62000 sentences for this evaluation although 19 millions sentences are available in Wikipedia. The major problem which we have faced for Odiya language is that standard corpus like CoNLL, BBN and MUC is not available in Odiya that is why we could not compare our result with the similar work [2, 7, and 9] which has already been done for English.

## 6. Conclusion and future work:

As the table no. 4 indicates, the result for Wikipedia corpus is outperforming over standard Odiya corpus. We have got a very promising result with a F-score of 78.89 for Odiya language. The accomplishment of our model may be boosted by creating bigger Wikipedia corpora, implementing more efficient inference techniques and word sense disambiguation. In this work, we have proved that Wikipedia is a great resource for creating annotated corpora for named entity recognition purpose but special care is needed for specific application. In time to come, we shall attempt to evaluate this system not only with standard Odiya corpus, rather with other languages like English as well. We will try to evaluate our system with CoNLL, BBN and MUC corpus, although the English language has so many features like capitalization etc for identifying NEs, which is not available for Odiya Language.

## References


[1] Sitanath Biswas. 2017. Hybrid Multilingual Named Entity Recognition for Indian Languages. International Journal of Control Theory and Application, 10 (18), 57-62.

[2] Joohui An, Seungwoo Lee, and Gary Geunbae Lee. 2003. Automatic acquisition of named entity tagged corpus from World Wide Web. In The Companion Volume to the Proceedings of 41st Annual Meeting of the Association for Computational Linguistics, pages 165–168.

[3] Agichtein, Eugene and Gravano, Luis. Snowball: 2000, Extracting relations from large plain-text collections. In In Proceedings of the 5th ACM International Conference on Digital Libraries, pages 85–94,. Asif Ekbal. Classifier Ensemble Selection Using Genetic Algorithm for Named Entity Recognition. Research on Language and Computation.Spriger.

[4] Chris.D.Paice, 1990, "An Evaluation method for Stemming algorithm," in Proceedings of the 17th Annual International ACM SIGR Conference on Research and Development in Information Retrieval, pp. 42–50.

[5] L. S.Larkey, M. E.Connel, and N. A. Jaleel, 2003, "Hindi CLIR in Thirty days," ACM Transaction on Asian language Information Processing, vol. 2(2), pp. 130–142.

[6] Wisam Dakka and Silviu Cucerzan. 2008. Augmenting Wikipedia with named entity tags. In Proceedings of the 3rd International Joint Conference on Natural Language Processing, pages 545–552, Hyderabad,India.

[7] Oren Etzioni, Michael Cafarella, Doug Downey, AnaMaria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. Artificial Intelligence, 165(1):91–134.

[8] Jun'ichi Kazama and Kentaro Torisawa. 2007. Exploiting Wikipedia as external knowledge for named entity recognition. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 698–707.

[9] Tibor Kiss and Jan Strunk. 2006. Unsupervised multilingual sentence boundary detection. Computational Linguistics, 32:485–525.

[10] Andrei Mikheev, Marc Moens, and Claire Grover. 1999. Named entity recognition without gazetteers. In Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics, pages 1–8, Bergen, Norway.

[11] Borthwick Andrew. 1999. A Maximum Entropy Approach to Named Entity Recognition. Ph.D. thesis, Computer Science Department, New York University.

[12] Fellbaum, C., 1998, editor. WordNet: An Electronic Lexical Database. MIT Press, Cambridge, MA.

[13] Kumarn. and Bhattacharyya Pushpak. 2006. Named Entity Recognition in Hindi using MEMM. In Technical Report, IIT Bombay, India.

[14] Nguyen, Dat P. T., Matsuo, Yutaka, and Ishizuka, Mitsuru. 2007, Relation extraction from wikipedia using subtree mining. In AAAI'07: Proceedings of the 22nd national conference on Artificial intelligence, pages 1414–1420. AAAI Press.

[15] Vapnik VN 1998, Statistical learning theory. Wiley, New York.

[16] Ada Brunstein. 2002. Annotation guidelines for answer types. LDC2005T33, Linguistic Data Consortium, Philadelphia.

[17] Razvan Bunescu and Marius Pas¸ca. 2006. Using encyclopedic knowledge for named entity disambiguation. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, pages 9–16.

[18] Ruiz-casado, Maria, Alfonseca, Enrique, and Castells, Pablo. 2005, Automatic assignment of wikipedia encyclopedic entries to wordnet synsets. In Proceedings of the Atlantic Web Intelligence Conference, AWIC-2005. Volume 3528 of Lecture Notes in Computer Science, pages 380–386. Springer Verlag.

[19] Massimiliano Ciaramita and Yasemin Altun. 2005. Named-entity recognition in novel domains with external lexical knowledge. In Proceedings of the NIPS Workshop on Advances in Structured Learning for Text and Speech Processing.

[20] Daniel Gildea. 2001. Corpus variation and parser performance. In 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP), Pittsburgh, PA.

[21] Suchanek, Fabian M., Kasneci, Gjergji, and Weikum, Gerhard. Yago: 2008, A large ontology from wikipedia and wordnet. Web Semant., 6(3):203–217.

[22] Tibor Kiss and Jan Strunk. 2006. Unsupervised multilingual sentence boundary detection. Computational Linguistics, 32(4):485–525.

[23] Chinchor, Nancy. 1995. MUC-6 Named Entity Task Definition (Version 2.1). In MUC-6. Maryland.

[24] Chinchor, Nancy. 1998. MUC-7 Named Entity Task Definition (Version 3.5). In MUC-7. Fairfax.

[25] Chikashi Nobata, Nigel Collier, and Jun'ichi Tsuji. 2000. Comparison between tagged corpora for the named entity task. In Proceedings of the Workshop on Comparing Corpora, pages 20–27.

[26] Moldovan, D., S. Harabagiu, R. Girju, P. Morarescu, F. Lacatusu, A. No-vischi, A. Badulescu, and O. Bolohan. 2002. LCC Tools for Question Answering. In Text REtrieval Conference (TREC) 2002.

[27] Babych, Bogdan and A. Hartley. 2003. Improving Machine Translation Quality with Automatic Named Entity Recognition.

In Proceedings of EAMT/EACL 2003 Workshop on MT and other Language Technology Tools, pages 1–8.
[28] Humphreys, K., R. Gaizauskas, S. Azzam, C. Huyck, B. Mitchell, H. Cun-nigham, and Y.Wilks. 1998. Univ. Of Sheffield: Description of the LaSIE-II System as Used for MUC-7. In MUC-7. Fairfax, Virginia.
[29] Aone, Chinatsu, L. Halverson, T. Hampton, and M. Ramos-Santacruz. 1998. SRA: Description of the IE2 system used for MUC-7. In MUC-7. Fairfax, Virginia.
[30] Mikheev, A., C. Grover, and M. Moens. 1998. Description of the LTG system used for MUC-7. In MUC-7. Fairfax, Virginia.
[31] Mikheev, A., C. Grover, and M. Moens. 1999. Named Entity Recognition without Gazeteers. In Proceedings of EACL, pages 1–8. Bergen, Norway.
[32] Joel Nothman, James R Curran, and Tara Murphy. 2008. Transforming Wikipedia into named entity training data. In Proceedings of the Australasian Language Technology Association Workshop 2008, pages 124–132, Hobart, Australia, December.
[33] Miller, S., M. Crystal, H. Fox, L. Ramshaw, R. Schawartz, R. Stone, R. Weischedel, and the Annotation Group. 1998. BBN: Description of the SIFT System as Used for MUC-7. In MUC-7. Fairfax, Virginia.
[34] Bikel, Daniel M., Richard L. Schwartz, and Ralph M. Weischedel. 1999. An Algorithm that Learns What's in a Name. Machine Learning 34(1-3):211– 231.
[35] Borthwick, A. 1999. Maximum Entropy Approach to Named Entity Recognition. Ph.D. thesis, New York University.
[36] Borthwick, Andrew, J. Sterling, E. Agichtein, and R. Grishman. 1998. NYU:Description of the MENE Named Entity System as Used in MUC-7. In MUC-7. Fairfax.
[37] Bennet, Scott W., C. Aone, and C. Lovell. 1997. Learning to Tag Multilingual Texts Through Observation. In Proceedings of Empirical Methods of Natural Language Processing, pages 109–116. Providence, Rhode Island.
[38] Joel Nothman, Tara Murphy, and James R. Curran. 2009. Analysing Wikipedia and gold-standard corpora for NER training. In Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), pages 612–620, Athens, Greece, March.
[39] Joel Nothman. 2008. Learning Named Entity Recognition from Wikipedia. Honours Thesis. School of IT, University of Sydney.
[40] PediaPress. 2007. mwlib Media Wiki parsing library. http://code.pediapress.com.
[41] Alexander E. Richman and Patrick Schone. 2008. Mining wiki resources for multilingual named entity recognition. In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 1– 9, Columbus, Ohio.
[42] Erik F. 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 on Natural Language Learning, pages 142–147, Edmonton, Canada.
[43] Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In Proceedings of the 6th Conference on Natural Language Learning, pages 1–4, Taipei, Taiwan.
[44] Ralph Weischedel and Ada Brunstein. 2005. BBN Pronoun Coreference and Entity Type Corpus. LDC2005T33, Linguistic Data Consortium, Philadelphia.
[45] FeiWu, Raphael Hoffmann, and Daniel S.Weld. 2008. Information extraction from Wikipedia: Moving down the long tail. In Proceedings of the 14th International Conference on Knowledge Discovery & Data Mining, Las Vegas, USA, August.
[46] A. Ramanathan and D.Rao, "A light weight stemmer for Hindi," in Proceedings of the 10th Conference of the European Chapter of the association for Computational Linguistic for South Asian language workshop, 2003, pp. 42–48.
[47] L. S.Larkey, M. E.Connel, and N. A. Jaleel, "Hindi CLIR in Thirty days," ACM Transaction on Asian language Information Processing, vol. 2(2), pp. 130–142, 2003.
[48] S. Dasgupta and V. Ng, "Unsupervised morphological parsing of Bengali," Language Resources and Evaluation, pp. 311–330, 2006.