EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

Instituto Politécnico Nacional
"La Técnica al Servicio de la Patria"

Research in Computing Science

ISSN: 1870-4069

Vol. 147 No. 9
September 2018

Research in Computing Science

Research in Computing Science, Año 17, Volumen 147, No. 9, septiembre de 2018, es una publicación mensual, editada por el Instituto Politécnico Nacional, a través del Centro de Investigación en Computación. Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, Ciudad de México, Tel. 57 29 60 00, ext. 56571. https://www.rcs.cic.ipn.mx. Editor responsable: Dr. Grigori Sidorov. Reserva de Derechos al Uso Exclusivo del Título No. 04-2019-082310242100-203. ISSN: en trámite, ambos otorgados por el Instituto Politécnico Nacional de Derecho de Autor. Responsable de la última actualización de este número: el Centro de Investigación en Computación, Dr. Grigori Sidorov, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738. Fecha de última modificación 01 de septiembre de 2018.

Las opiniones expresadas por los autores no necesariamente reflejan la postura del editor de la publicación.

Research in Computing Science, year 17, Volume 147, No. 9, September 2018, is published monthly by the Center for Computing Research of IPN.

The opinions expressed by the authors does not necessarily reflect the editor's posture.

Advances in Computational Linguistics

Alexander Gelbukh (ed.)

SEP
SECRETARÍA DE
EDUCACIÓN PÚBLICA

Instituto Politécnico Nacional
“La Técnica al Servicio de la Patria”

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2018

ISSN: 1870-4069

Formerly ISSN: 1665-9899

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional “Adolfo López Mateos”, Zacatenco
07738, México D.F., México

http://www.rcs.cic.ipn.mx
http://www.ipn.mx
http://www.cic.ipn.mx

The editors and the publisher of this journal have made their best effort in preparing this special issue, but make no warranty of any kind, expressed or implied, with regard to the information contained in this volume.

Indexed in LATINDEX, DBLP and Periodica

Electronic edition

ISSN 1870-4069

Table of Contents

Page

Using Information Extraction and Search Engines for Automatic Detection of Inadequate Descriptions and Information Supplements in Japanese Wikipedia 5
Masaki Murata, Naoya Nonami, Qing Ma

Object Recognition Approach based on Color Distribution 19
Nada Farhani, Naim Terbeh, Mounir Zrigui

Machine Translation Evaluation Metrics for Recognizing Textual Entailment 31
Tanik Saikh, Asif Ekbal, Debajyoty Banik, Pushpak Bhattacharyya

Classification and Annotation of Arabic Factoid Questions: Application to Medical Domain 45
Essia Bessaies, Slim Mesfar, Henda ben Ghezala

A Rule-Based Approach for Arabic Named Entity Metonymy Resolution 53
Sadek Mansouri, Chahira Lhioui, Mounir Zrigui, Mbarek Charhad

Customer Feedback Analysis Using Machine Learning Techniques 63
Anupam Jamatia, Kanishka Joshi, Kundan Kumar, Shivam Kumar, Subrito Haldar

Opinion Summarization Using Product Feature Based Review Classification 73
K Vimal Kumar, Pawan Upadhyay, Abheet Kumar Gupta, Deepanshu Wadhwa

Translation of Wikipedia Category Names by Pattern Analysis 85
Thang Ta Hoang, Alexander Gelbukh

ISSN 1870-4069

Using Information Extraction and Search Engines for Automatic Detection of Inadequate Descriptions and Information Supplements in Japanese Wikipedia

Masaki Murata[1,2], Naoya Nonami[2], Qing Ma[3]

[1] Tottori University, Cross-informatics Research Center,
Japan

[2] Tottori University, Faculty of Engineering,
Japan

[3] Ryukoku University, Faculty of Advanced Science and Technology,
Seta, Otsu, Japan

{murata,s132043}@ike.tottori-u.ac.jp,
qma@math.ryukoku.ac.jp

Abstract. When sentences lack important information that the reader wishes to know, they are difficult to read. Thus, correcting inadequate sentences is easier if there is a technique for pointing out inadequate descriptions and adding the missing information. Akano et al. [1] constructed such a technique to detect inadequate descriptions using information extraction; however, the technique did not fill in the missing information. Therefore, this study was conducted, wherein a web search engine was used to provide appropriate information to inadequate sentences. In our method, a search engine gathers webpages and extracts important information from them. In our experiments, the MRR (mean reciprocal rank) and top-five accuracy of measuring important information for inadequate sentences were 0.77 and 0.96, respectively, under a certain experimental condition, which eliminated cases wherein correct answers were not found in webpages. The f-measure for this very difficult task of detecting inadequate descriptions and correct expressions in webpages was 0.77, which indicates that our method was very effective.

Keywords: Inadequate description · Information supplement · Web search engine · Japanese Wikipedia · information extraction.

1 Introduction

When sentences lack important information that the reader wishes to know, they are difficult to read. Thus, correcting inadequate sentences is easier if there is a technique for pointing out inadequate descriptions and adding the missing information.

This study treated items that commonly appear in many entries in the Japanese Wikipedia as important items and information from these entries were extracted, arranged, and displayed in the form of a table. Any blank cells in the table are set as inadequate descriptions wherein important information is missing. Important information is specific information that is included in the important items of the table. For example, in Table 1, the important items, "personal name" and "organization name," are missing parts that are not described in the corresponding Wikipedia entry on a castle. By pointing out these missing parts, we can modify the adequate descriptions easily. Although it is useful to point out that there are missing parts in the table, it is more helpful to correct and complement the adequate descriptions if we can fill in the missing parts of the table, as shown in Table 2. The words in parentheses in Table 2 indicate complementary information.

Therefore, in this study, we used a search engine to fill in the appropriate information for the missing parts. In this study, supporting the correction of sentences by filling in the missing parts with the appropriate information corresponds to text correction support. Referring to the information supplements for the missing parts is essential while correcting inadequate sentences with missing parts.

Table 1. Blanks in the table correspond to missing parts (inadequate descriptions).

	Location	Person	Organization
Uwajima Castle	Uwajima		

Table 2. Missing parts in the table that are filled in.

	Location	Person	Organization
Uwajima Castle	Uwajima	(Takatora Todo)	(Uwajima clan)

This study has the following characteristics.

- Originality: using information extraction to detect inadequate descriptions in Wikipedia and using a search engine to complement inadequate descriptions. Our purpose is to detect inadequate descriptions in Wikipedia and correct them with a search engine. We extract important information from Wikipedia by extracting clustered words and compiling the extracted information into a table. A blank cell in the table is considered as a missing part. We complement the missing parts using webpages found with the search engine. Our method's originality lies in our complementing the missing parts in addition to detecting them. Although a previous paper [1] also focused on the detection of missing parts, it did not complement them.

- Information extraction from webpages.
  Fifty items were acquired using the search engine and important information was extracted from the webpages by clustering. We compiled the extracted information into a table. The top-five accuracy of the extracted information was 0.66.
- Supplementation of missing parts.
  We checked how accurately we complemented the missing parts of the table. In the experiments, the MRR and top-five accuracy for showing important information for inadequate sentences were 0.77 and 0.96, respectively, in eliminating cases wherein the correct answers were not in the webpages. The f-measure for the very difficult task of detecting inadequate descriptions and detecting correct expressions in the webpages was 0.77; therefore, our method was very effective.

2 Related Works

Akano et al. [1] extracted important information using clustered words and summarized it in a table. Blank spots in the table were considered as missing parts (inadequate descriptions). The user was informed of these missing parts and asked to add descriptions to support the writing preparation. Although the previous study pointed out the missing parts in a table, there has been no study on a method for asking the user to fill in the missing information. Therefore, in this study, we used a search engine to acquire the information to fill in the missing parts and support text correction.

Murata et al. [4] improved the accuracy of the question answering system by a method of using scores in multiple documents. Their study and our study are similar in that they both detect answer expressions using document retrieval but Murata et al. focused on the question-answering system, whereas we handled the writing preparation support.

Okada et al. [7] handled text preparation support by the automatic detection of information that should be present in a paper. They defined the information, such as research results, effectiveness, and necessity, that should be described in papers as items requiring mention (IRM). Okada et al. provided sentence creation support by automatically detecting the missing IRMs. In their study, they supported writing by automatically detecting whether important items were missing. Their system is similar to ours in the use of important items for the detection of inadequate descriptions that lack important information. However, the studies differ in that their research focused on sentences in an academic paper, whereas our research focused on sentences in Wikipedia. Unlike the system of Okada et al., our system implements word clustering for using important items and complementing missing parts.

Fukuda et al. [2] constructed a system that extracts expressions, indicating the effects and trends of technology from research papers, and compiled the extracted information. Ptaszynski et al. [6] developed a system to support the writing of research papers by preparing data and automatically conducting the experiments to obtain accuracies. Using LaTeX templates, the system creates tables containing all the results and graphs these results. Their study is useful for the surveying and preparation of research papers. However, these two systems were not intended to support the correction of inadequate portions of sentences in a document, whereas our system provides support for correcting inadequate descriptions in Wikipedia.

A study by Nadamoto et al. proposed a technique that recognizes missing parts [5]. They observed that discussions in community-type content, such as social networking services or blogs, may concentrate on a small domain, and thereby, miss some viewpoints. They have termed the missed viewpoints as *content holes* and proposed a method of detecting such occurrences by comparing discussions in community-type content and general information such as the sort of material that appears in Wikipedia. Regarding the detection of missing information, the study of Nadamoto et al. and our study are similar but their study focused on community-type contents, whereas ours focused on descriptions in a document.

Tsudo et al. [8] conducted a study of the automatic detection of redundant sentences to support the writing of sentences. They proposed a method of using machine learning to detect redundant sentences automatically. Many other systems support the writing of sentences, but other than those by Akano et al. and Okada et al., few detect inadequate descriptions with important items.

3 Proposed Method

Our method detects inadequate descriptions in documents (i.e. Wikipedia pages) and extracts expressions useful for complementing the inadequate descriptions using search engines.

The method proposed by this study comprises two stages: (i) extraction of important information from documents (Wikipedia pages), and (ii) information extraction using search engines.

Using the clustering tool in word2vec [3], we extract important items from entries in the Japanese Wikipedia about castles.

3.1 Extraction of Important Information from Documents (Wikipedia Pages)

The method described here is based on that of Akano et al. [1].

We extract important items from the entries in Wikipedia about castles using the clustering tool in word2vec. Extraction is done for each entry.

We use “word clustering” in word2vec to select important items related to the extracted data. Word clusters are created by grouping highly similar words. We assign a number to each cluster, select the important items manually, and compile the items into a table. The method of compiling the information into a table is described below.

1. We determine what we want to extract. We extract pages containing things we determine from Wikipedia.
2. Using the clustering function of word2vec, we cluster words in the extracted data. A number is assigned to each cluster. Similar word groups are put into clusters. For example, word groups of locations and personal names are grouped into Clusters No. 1 and 2, respectively. Examples are shown in Tables 3 and 4.
3. A cluster of words corresponds to a column and a page of extracted data corresponds to a row in a table. We place a word belonging to a cluster appearing on a page into the corresponding row and column positions. If more than one word of a cluster appears on a page, we fill all those words in that place of the table.
4. We find the total number of words (called Frequency A) in each column of the table. We sort the columns in the table so that a column with a higher Frequency A is on the left. We delete the columns of clusters with less Frequency A.
5. We manually select columns (important items) that are considered to be important information on castles from the columns of the cluster numbers with high Frequency A according to the sorted table. We delete the unselected columns and make a table. Table 5 is an example of a table created in this way.

   For example, we assume that the entry on Osaka Castle contains the following sentences:

   > *oosaka jo wa oosaka ni shozai suru. toyotomi hideyoshi ga kizuita oosaka jo no ikou wa, genzai subete maibotsu shiteiru.*
   >
   > (Osaka Castle is located in Osaka. The remains of Osaka Castle, built by Toyotomi Hideyoshi, are now all buried.)

   Because the location name, Osaka, and the personal name, Toyotomi Hideyoshi, appear on the page, “Osaka” is extracted and placed into a word cluster of location names and “Toyotomi Hideyoshi” is extracted and placed into a word cluster of personal names. A table such as Table 5 is created.

Table 3. Words corresponding to locations (Cluster No. 1).

Location
Kyoto
Osaka
Miyagi

3.2 Information Extraction Using Search Engine

A blank in the table created by the method described in Section 3.1 indicates that the corresponding page does not contain important information. Therefore,

Table 4. Words corresponding to personal names (Cluster No. 2).

Name
Masamune Date
Ieyasu Tokugawa
Hideyoshi Toyotomi

Table 5. Arranged table.

Castle	Location	Person
Osaka Castle	Osaka	Hideyoshi Toyotomi
Nijo Castle	Kyoto	Ieyasu Tokugawa
Sendai Castle	Miyagi	Masamune Date

we acquire information to fill in the blank using the search engine to complement the table.

We first input the name of a castle as a query to the search engine, which returned 50 webpages. We extracted important information from these using the method described in Section 3.1. Among the extracted information, the five most frequent words were compiled into a table, which was presented to a user to correct the sentences.

An example of extracting important information from webpages and compiling it in a table is shown in Tables 6 and 7.

Table 6 is constructed by the method of extracting information from Wikipedia pages, as described in Section 3.1. In the Wikipedia entry for Nezoe Castle, no information exists for the age and era name. Therefore, the columns for the age and era name of Nezoe Castle are left blank to indicate the missing parts.

Table 6. Table with missing parts provided by extracting information from Wikipedia.

Castle	Prefecture	Age	Location	Era name
Nezoe Castle	Miyagi		Sendai	

Table 7. Table complementing missing parts provided by extracting information from webpages.

Castle	Prefecture	Age	Location	Era name
Nezoe Castle	Miyagi	(Heian age)	Sendai	(Eisho)

We construct Table 7 from Table 6 by using a method for extracting information from webpages. We input “Nezoe Castle” as a query into the search engine, which extracts "prefecture", "age," "location," and "era name" from the returned webpages and compile the information into a table.

The following sentences are excerpts from one page randomly selected from the 50 returned pages:

> *nezoe jo wa heian jidai ni sakaeta abeshi no yakata to sareru. eisho* 6 *nen* (1051) *ni hajimaru zenkunen no ekide, genji no gunzei ga nezoe jo wo kougekishita.*
> (Nezoe Castle was a castle used by Mr. Abe, who flourished during the Heian age. Genji's army attacked Nezoe Castle during the Zenkunen War in the sixth year of Eisho (A.D. 1051).)

Because "Heian age," which is the name of an age, and "Eisho," which is the name of an era, appear in the document, these words are extracted and output to the table. When comparing Tables 6 and 7, Table 7 can contain "age" and "era name," whereas Table 6 cannot do so. Since important information not described in Wikipedia can be acquired from webpages in this way, the method can be used for supporting the correction of sentences.

4 Experiments

4.1 Experimental Conditions

For this study, from the Japanese Wikipedia (November 2014), we used 2,665 pages whose titles ended in "castle."

We input the pages and extracted the information using the method described in Section 3.1 and created a table for detecting incomplete descriptions. We used the results of Akano et al. [1].

In the experiments, Clusters 401, 407, and 765 were selected manually from the clustering of the important items. The rows of the table listed the names of the castle and the columns listed the important items. Cluster 401 contained information on battles, 407 contained information on the construction of the castles, and 765 contained information related to traffic.

Using the method described in Section 3.2, we created a table to facilitate the correction of incomplete descriptions in the Wikipedia entries. To evaluate the method, we conducted the following two experiments:

- Information extraction using search engines for all parts of the table.
- Information extraction using search engines for only missing parts of the table.

The experiment for using search engines to extract information from the webpages for all parts of the table was conducted to observe the performance of the information extraction.

Information extraction and table creation were performed on the 2,665 pages in Wikipedia about castles (November 2014) using the methods described in Section 3.1. In addition, we input 30 names of castles randomly selected from the 2,665 pages into the search engine, acquired the webpages, and conducted the experiment with these 30 names to evaluate the method. The search engine used was Microsoft's Bing Search API.

4.2 Evaluation Method

Using top-one accuracy, top-five accuracy, and MRR, we conducted an evaluation. In top n accuracy, the output is judged to be correct when the top n candidate answers contain a correct answer. In MRR, a score of $1/r$ is given when the r-th candidate answer is correct.

4.3 Experimental Results of Information Extraction Using Search Engines for all Parts of the Table

Using the method proposed in Section 3.2, we entered 30 names of castles randomly selected from the 2,665 pages about castles in Wikipedia (November 2014) into the search engine. Using the webpages acquired from the search engine, a table was created by the method described in Section 3.2.

An example of the results is shown in Table 8. The words in bold font were those judged to be correct. The five most frequent words in the 50 webpages acquired by the search engine were output for each important item. Table 9 shows the results of the top-one accuracy, the top-five accuracy, and MRR evaluations of the constructed table.

The top-five accuracy was 0.66. We found that we could extract information using webpages to a certain extent.

4.4 Experimental Results of Information Extraction Using Search Engines for only Missing Parts of the Table

In the experiments extracting important information from documents (Wikipedia pages) described in Section 3.1, we detected the missing parts by consulting Akano et. al.'s paper [1]. The method detected 67 missing parts. There were 55 correct missing parts in the data of the 30 castles used in the study. Among the correct missing parts, 53 had been correctly detected. The f-measure, precision, and recall of extracting the correct missing parts are shown in Table 10. The f-measure was 0.84, indicating a very good result.

For the information extraction using a search engine, as described in Section 3.2, we used our method in the experiments.

We conducted experiments to extract information using search engines for only the missing parts of the table with no correct answers in Wikipedia. The results are shown in Table 11. The experiment was performed on the missing parts that had been correctly detected by the method described in Section 3.1. "No correct answers in Wikipedia" means the case where a description in Wikipedia is inadequate and we should complement the inadequate description.

The top-five accuracy of the experiments on the missing parts of the table with no correct answers in Wikipedia was 0.50.

Table 8. Example of results of information extraction.

	Cluster 401 (battle)	Cluster 407 (construction of castle)	Cluster 765 (traffic)
Uwajima Castle	**opening a castle** attack falling castle immediately sortie	**main castle keep** Nagaya gate relocation palace remains of gate	**traffic** Kaido highway Setouchi Hokuriku
Chikugo Jugo Castle	**great defeat** resistance outing falling castle fierce	office main castle keep second **government office** palace	**Kaido** traffic contact tie highway
Okazaki Castle	**departure** break return great defeat battle field	**main castle keep** second castle keep main gate palace government office	**traffic** Kaido Tokaido Hokuriku convenience
Sakurao Castle	**falling castle** opening a castle outing occupation Return	**main castle keep** second castle keep rising sun office Gate	**Setouchi** pilgrimage Sanyo traffic convenience
Linderhof Castle	return arson immediately a few	transfer rising sun	**traffic** highway convenience Kaido romantic
Odawara Castle	**opening a castle** falling castle retreat return withdrawal	**main castle keep** second castle keep main gate palace lotus pond	**Kaido** Tokaido traffic trunk line Hokuriku
Kawata Castle	**falling castle** approach run surrender ambush	**main castle keep** second castle keep palace office address	**highway** traffic strategic stop convenience crossing
Nagamori Castle	falling castle **going out** offend opening a castle withdrawal	main castle keep second castle keep **palace** great gate office	**traffic** Kaido Nakasendo Hokuriku convenience
Shakujii Castle	**falling castle** defeat arson return great defeats	**main castle keep** Nagaya gate office second house keep great gate	**contact** convenient traffic Kaido highway

Table 9. Accuracies in all the parts.

Evaluation method	Cluster 401	Cluster 407	Cluster 765	Total
Top-1 Accuracy	0.56 (17/30)	0.50 (15/30)	0.53 (16/30)	0.53 (48/90)
Top-5 Accuracy	0.70 (21/30)	0.60 (18/30)	0.70 (21/30)	0.66 (60/90)
MRR	0.62	0.52	0.57	0.55

Table 10. F-measure of detecting inadequate descriptions (missing parts).

Precision	0.79 (53/67)
Recall	0.96 (53/55)
F-measure	0.84

Table 11. Accuracies in missing parts of the table with no correct answers in Wikipedia.

Evaluation method	Cluster 401	Cluster 407	Cluster 765	Total
Top-1 Accuracy	0.35 (5/14)	0.29 (5/17)	0.36 (8/22)	0.33 (18/53)
Top-5 Accuracy	0.50 (7/14)	0.41 (7/17)	0.59 (13/22)	0.50 (27/53)
MRR	0.42	0.32	0.45	0.40

Table 12. Accuracies in the missing parts of the table with no correct answers in Wikipedia (Condition A: eliminating cases when no correct answers exist in the webpages).

Evaluation method	Cluster 401	Cluster 407	Cluster 765	Total
Top-1 Accuracy	0.71 (5/ 7)	0.62 (5/ 8)	0.61 (8/13)	0.64 (18/28)
Top-5 Accuracy	1.00 (7/ 7)	0.87 (7/ 8)	1.00 (13/13)	0.96 (27/28)
MRR	0.85	0.69	0.77	0.77

Table 12 is the result of the evaluation we conducted under the following Condition A.

When the correct answer was not in the answer candidates within the first to fifth ranks, we confirmed if the correct answer was in the answer candidates within the sixth to twentieth ranks. If the correct answer was not in the answer candidates, we manually confirmed whether the correct answer was on the Web. If the correct answer was not on the Web, the missing part was the part for which we did not have to find correct answers. We conducted experiments by eliminating such missing parts.

Under Condition A (eliminating cases when no correct answer exists in the webpages), the MRR and top-five accuracy for the experiments in all the parts of the table with no correct answers in Wikipedia were 0.77 and 0.96, respectively. The results were very accurate, indicating that with high accuracy, our method could gather information useful for complementing the missing parts when correct answers existed in webpages.

We calculated the f-measures for detecting inadequate descriptions and correct expressions in webpages. The results are shown in Table 13. The recall rate is the rate of dividing the number of detected correct missing parts and outputting correct answers with no correct answers in Wikipedia by the number of correct answers in the case of using the top-five accuracy. The precision rate is obtained by dividing the number of detected correct missing parts and outputted correct answers with no correct answers in Wikipedia by the number of detected missing parts and output answers with no correct answers in Wikipedia in the case of using the top-five accuracy.

The task of detecting inadequate descriptions and detecting correct expressions in webpages is very difficult. Therefore, our method was very effective since it obtained an f-measure of 0.77.

Table 13. F-measure for detecting inadequate descriptions and detecting correct expressions in webpages

Precision	0.68 (27/40)
Recall	0.90 (27/30)
F-measure	0.77

We show an example of the successful results of our method. Using the method described in Section 3.1, Table 14 was created by extracting important information from an entry in Wikipedia. Blanks in the tables are missing parts that correspond to the absence of descriptions of correct answers in Wikipedia. Therefore, the blanks are correct. For Table 14, we used the method proposed in Section 3.2 to complement the missing parts in this table. Table 15 shows what had actually been complemented. The words in parentheses are complementary information, which includes the information taken from the first to fifth candidate answers.

Table 14. Examples of a table having missing parts.

	Cluster 401 (battle)	Cluster 407 (construction of castle)	Cluster 765 (traffic)
Anozu Castle			
Moji Castle	**defeat**	**castle keep**	

Table 15. Successful examples of sentence correction support.

	Cluster 401 (battle)	Cluster 407 (construction of castle)	Cluster 765 (traffic)
Anozu Castle	(**opening a castle**)	(**castle keep**)	(**highway**)
Moji Castle	**defeat**	**castle keep**	(**traffic**)

We give an example of the failures of our method. Our method could not obtain the correct answers in the top-five candidates for Cluster 401 (battles) for Tabata Castle. The correct answer, "great defeat," appears as the 16th candidate in our method. Therefore, the case was judged to be incorrect for top-five accuracy. Table 16 shows 20 candidate answers for Cluster 401 (battles) in Tabata Castle. Although "departure," "reinforcement," "falling castle," etc. in the first to third ranks may seem to be correct answers, they appear in a context unrelated to Tabata Castle and so they are incorrect.

Table 16. Words of Cluster 401 with the 1st to 20th ranks for Tabata Castle.

Castle	Rank	Cluster 401 (battles)	Number of pages
Tabata Castle	1	departure	4
	2	reinforcement	3
	3	falling castle	2
	4	rushing	2
	5	immediately	2
	6	retreat	2
	7	joining	2
	8	surrender	1
	9	defense	1
	10	arson	1
	11	weapons	1
	12	dispatch	1
	13	rally	1
	14	stationed	1
	15	departure	1
	16	**great defense**	1
	17	annihilation	1
	18	burned down	1
	19	sortie	1
	20	heading	1

5 Conclusion

The purpose of this study is to support the correction of inadequate descriptions. Important information is extracted from entries in Wikipedia and inadequate descriptions are detected. Information that is useful for correcting the inadequate descriptions is gathered by a search engine and presented to a user to support the correction of the sentences.

In the experiments on the extraction of information using search engines for all parts of a table, our method obtained a top-five accuracy of 0.66. We found that we could extract information using Web pages to a certain extent.

In the experiments on the missing parts of the table with no correct answers in Wikipedia, the top-five accuracy was 0.50. Under Condition A (eliminating cases when no correct answers exist in webpages), experiments on the missing parts of the table with no correct answers in Wikipedia, obtained an MRR and a top-five accuracy of 0.77 and 0.96, respectively, indicating that the results were very accurate, and that our method could gather, with high accuracy, information to complement the missing parts when correct answers existed in webpages.

The f-measure of the very difficult task of detecting inadequate descriptions and detecting correct expressions in webpages was 0.68, indicating that our method was very effective, i.e., it obtained 0.77 in f-measure.

Acknowledgment. This work was supported by JSPS KAKENHI Grant Number 26330252.

References

1. Akano, H., Murata, M., Ma, Q.: Detection of inadequate descriptions in Wikipedia using information extraction based on word clustering. In: Proceedings of IFSA-SCIS 2017. pp. 1–6 (2017)
2. Fukuda, S., Nanba, H., Takezawa, T.: Extraction and visualization of technical trend information from research papers and patents. D-Lib Magazine (2012)
3. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems 26, 3111–3119 (2013)
4. Murata, M., Utiyama, M., Isahara, H.: Use of multiple documents as evidence with decreased adding in a Japanese question-answering system. Journal of Natural Language Processing 12(2), 209–248 (2005)
5. Nadamoto, A., Aramaki, E., Abekawa, T., Murakami, Y.: Extracting content-holes by comparing community-type content with wikipedia. The International Journal of Web Information Systems 6(3), 248–260 (2010)
6. Ptaszynski, M., Masui, F.: Spass: A scientific paper writing support system. In: The Third International Conference on Informatics Engineering and Information Science (ICIEIS2014). pp. 1–10 (2014)
7. Takuma Okada, M.M., Ma, Q.: Automatic detection and manual analysis of inadequate descriptions in a thesis. In: Proceedings of SCIS-ISIS 2016. pp. 916–921 (2016)
8. Tsudo, S., Murata, M., Tokuhisa, M., Ma, Q.: Machine learning for analysis and detection of redundant sentences toward development of writing support systems. In: Proceedings of the 13th International Symposium on Advanced Intelligent Systems. pp. 2225–2228 (2012)

ISSN 1870-4069

Object Recognition Approach based on Color Distribution

Nada Farhani[1,2], Naim Terbeh[2], Mounir Zrigui[2]

[1] University of Sousse, Hammam Sousse,
Tunisia

[2] University of Monastir, Monastir,
Tunisia

farhaninada@yahoo.fr, naim.terbeh@gmail.com,
mounir.zrigui@fsm.rnu.tn

Abstract. Until now, object recognition presents a difficult task and still requires research to improve the recognition results. The observation of an object can be very different depending on the field of activity as well as the reason to recognize and classify, perhaps simple or complex, and the tools developed are adapted to each use. It is in this context that this paper intervenes. We propose a recognition approach based on the color distribution using the histogram and spatiogram calculation, in the RGB space. With the used database, our approach gave good results. The main purpose of our work is to use the concepts from recognition to generate sentences in Arabic that summarize the content of the image.

Keywords: Object recognition, color distribution, histogram, spatiogram, learning.

1 Introduction

Recognition of objects is a delicate problem that takes place at the top level in the hierarchy of vision tasks and constitutes the most difficult computational part. To overcome this difficulty, a vision system must be able to combine its internal representational capacities in order to make successful decisions. Many difficulties appear in the recognition of objects, in particular those linked to the variability of appearance linked to light, orientation, etc. Most theories of object recognition only deal with the geometric aspect. Today we can count two large groups of theories, which diverge, on the format of the representation according to whether this one is independent or dependent on the views of the object to be represented.

The first group of these theories considers that the representation of an object is conceived as a set of characteristics (invariants) of the object which are independent of

pp. 19–29; rec. 2018-01-02; acc. 2018-03-05

the views of this same object [1]. This is a structural description of the object. One of the most interesting approaches is that of Biederman: "Recognition by Components" [2] which consists in representing the object by breaking it down into structures (primitives) according to a scheme proposed by de Marr and Nishihara [3].

The second group of these theories considers that the representation of an object is linked to views specific to the object and that any other view can be deduced using these views [4,5]. The models of this type of representation consider a view as a collection of characteristics (2D information, 3D information...). Recognition is expressed as a function of the images already seen. The model claimed by the first group can appear attractive thanks to a compact and robust representation for the objects.

However, experience has shown that it is very difficult to detect invariants in images on the one hand and that this structure of invariants necessarily leads to categorization and not to identification on the other. To remedy these drawbacks, new approaches have surfaced according to the theory of the second group. These approaches model objects by their images themselves, thus abandoning models of geometric type or based on invariant structures. Thus, an object is represented by a collection of images and the recognition is based on the matching of a new image of the object with the images in the collection.

Several works [4] have evolved in this direction. This paper is organized as follows. In the second section, an overview of the image description methods that help in object recognition tasks. In the third section, a state of art is presented. The fourth section presents the proposed approach based on the histogram and spatiogram calculation. Some experimental results as well as the discussion will be presented in section five. Finally, section six concludes this paper.

2 Image Description Methods

After doing image preprocessing, we are interested in the task of object recognition. In fact, it is necessary to extract areas of interest from information relating to what is contained in the image; for this, there are image descriptors that characterize the information available. Two ways of proceeding are possible:

- Either the image is described at specific and significant points: it is a local descriptor, and these interesting points are points of interest.
- Or all the pixels of the image corresponding to the area of interest are considered in the description: it is a global descriptor.

2.1 Local Descriptors

For local descriptors present methods that are based on correlation measures. They make it possible to quantify the resemblance between two pixels and their neighborhoods. There are methods to detect points of interest. For example, the Harris point detector [6], which is certainly the most widely used; its principle is to detect sudden changes in intensity in the image, thereby highlighting corners. Then a descriptor characterizes each point of interest by its neighborhood. Most often, this is the information the detector has calculated. Local methods are the most used today for the object recognition thanks to their good management of occultation, charged background and changes of point of view as well as their speed.

However, unfortunately, local methods do not take the entire image into account and are fraught with ambiguities. This results in many matching errors.

2.2 Global Descriptors

In object recognition, a global descriptor is easier to use because it processes the image in its entirety. The descriptor is thus less sensitive to distortions from one image to another. Two close images must therefore lead to two close descriptors. A classifier can thus be trained on this data during training, and it will be able to associate these two images with the same class thanks to a notion of distance between the descriptors (for example Euclidean distance).

The same will be true for recognition: by using the same descriptor as during training, the classifier will be able to associate a new object with a class of learned objects. Many methods exist. The choice depends above all on the intended application and the calculation time available to carry it out. Indeed, in the case of real-time recognition and detection, there is often a trade-off between the quality of the response and the execution time.

However, if the information provided by the descriptor is not precise enough, the classifier will be difficult to train during learning and the results it will provide in recognition will therefore be unreliable. Global methods take the entire image into account. They are based on the following principle: if the calculated disparity map is correct and if an image is constructed from the reference image and the disparity map, then the resulting image should resemble the other image. We then seek to find the disparity map which maximizes a global similarity function.

An example of a method is the method presented in [7]. This method is one of the best classified in the protocol of Scharstein and Szeliski1 of [8]. It is divided into four stages:

- Color segmentation.
- Use of an adaptive correlation score that maximizes the number of reliable matches.
- Assigning a disparity value to each region.
- Search for the optimal disparity using a belief propagation based on the Markov field model.

3 State of the Art: Objects Recognition

The authors, in [9], presented a residual learning framework in order to simplify the formation of deeper networks than previously used networks. In an explicit way, Zhang

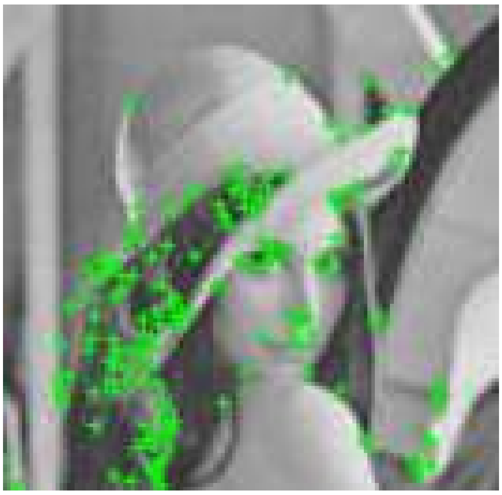


Fig. 1. Harris Detector - In order to find points of interest, the Harris Detector calculates, for each pixel, the autocorrelation matrix from the two components of the image's gradient vectors. Then, the detector response matrix is obtained from these matrices. Finally, the points of interest, here marked with a green cross, are located from this answer.

et al. did not choose to learn unreferenced functions, but they reformulated the layers as residual learning functions with reference to the layer inputs. In their work [10], Chabot & al. proposed a system that is divided into two stages. In fact, this system is used to transmit the input image via the Deep MANTA network which produces the visibility properties of the parts, the 2D bounding boxes and the associated vehicle geometry.

OverFeat [11] is an example of the first modern objects detectors, which is a one-stage detector based on deep networks. One of the newest detectors is SSD (Single Shot MultiBox Detector) [12], where its approach is based on a convolutional feed-forward network generating scores for the presence of object class instances. In fixed size bounding boxes that were generated before, then a non-maximal deletion step to produce the final detections. YOLO9000 [13] also a real-time framework is designed to detect more than 9000 categories of objects while optimizing the classification and detection. In order to detect faces, in their works [14], the authors exploited detectors of boosted objects.

The use of integral channels [15] and HOG functions [16] has led to the emergence of effective methods for pedestrian detection. In the classical computer vision, the sliding window approach was the main model of detection, with the appearance of deep learning [17]. With regard to work based on two-stage detectors, in [18], the first phase is the generation of an isolated set of candidate propositions in the obligation to include all the objects as well as the filtering of the majority negative locations. The second phase classifies proposals into foreground / background.

For the purpose of improving the second-stage classifier, R-CNN [19] has made modifications to this level to have a convolutional network that offers significant gains

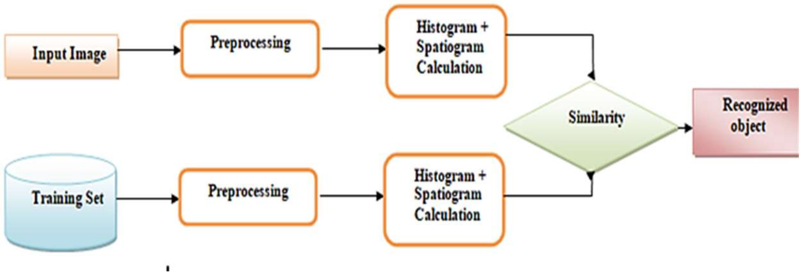


Fig. 2. Recognition steps. This figure shows the details of the steps to follow in order to recognize object (s) in an image.

in accuracy. In turn, R-CNN has also been improved over the years, at the same time at the speed level [20] and by exploiting proposals for learned objects [21]. To improve timeliness and to have a faster classifier than that used by the RCNN, Region Proposal Networks (RPPs) incorporated proposal generation with the second-stage classifier into a single convolutional network [21]. As well as several extensions have been proposed in this framework [22, 23, 24]. In [25], Karpathy & al. proposed a model generating natural language descriptions of images and their regions.

To learn more about the correspondence between images and language, the approach proposed the authors exploits the dataset of text descriptions of images. The base of the alignment model is a new combination of two-way recursive neural networks on sentences, Convolutional neural networks on image regions, and a structured objective that serves to align the two modalities via multimodal integration.

The approach proposed in [26] by Lin & al. is the focal loss applying a modulating term to the loss of cross entropy in order to weight the many easy negatives and to focus learning on concrete examples. Vaillant & al. applied in their work [27] convolutional neural networks to the recognition of handwritten figures. Many researchers work on the detection in different fields [28, 29, 30, 31, 32], but the objects detection and recognition still a challenge in the field of research because of several difficulties that the researcher can envisage because of the variability of shape, position, contrast of objects.

4 The Proposed Approach

Since we want to build an image description system that can be used by humans, and it is the human being who gives meaning to what he sees, it may be interesting to draw inspiration from of the human perception system to choose the visual spaces in order to come as close as possible to the human being's understanding of the image, and thus reduce the semantic gap. Low-level descriptors are used at the level of global methods, also called feature vectors, such as color, texture, and shape.

In our approach we will use the color distribution in order to make the objects recognition task which is based on the histogram and spatiogram calculation. After

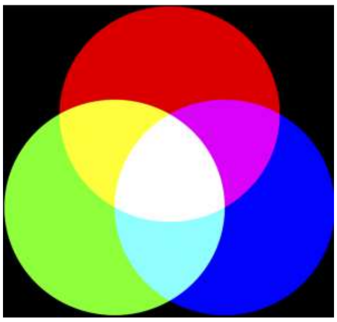


Fig. 3. Additive synthesis of colors.

building our database, we will do the preprocessing step for the images, after this the training step is done.

For the test phase, also we do a pretreatment for the image, calculate the histogram and spatiogram. And for the recognition task we will opt for a similarity calculation to find the recognized object. The following figure presents the system of objects recognition.

4.1 Color Image

A color image is actually made up of three images, in order to represent red, green, and blue. Each of these three images is called a channel. This representation in red, green and blue reflects the functioning of the human visual system.

Each pixel in the color image thus contains three numbers (r, g, b), each being an integer between 0 and 255. If the pixel is equal to (r, g, b) = (255, 0, 0), it contains only red information, and is displayed as red. Similarly, pixels of (0, 255, 0) and (0, 0, 255) are displayed green and blue, respectively. A color image can be displayed on the screen from its three channels (r, g, b) using the rules of additive color synthesis. The following figure shows the composition rules for this additive synthesis of colors. A pixel with the values (r, v, b) = (255, 0, 255) is a mixture of red and green, so it is displayed as yellow. Figure 4 shows the decomposition of a color image into its three constituent channels.

4.1.1 Color Image Histogram

For a monochrome image, that is to say with a single component, the histogram is defined as a discrete function which associates with each intensity value the number of pixels taking this value. The histogram is therefore determined by counting the number of pixels for each intensity of the image. Sometimes a quantization is carried out, which groups together several intensity values in a single class, which can make it possible to better visualize the distribution of the intensities of the image. Histograms are usually normalized by dividing the values of each class by the total number of pixels in the image. The value of a class then varies between 0 and 1 and can be interpreted as the probability of occurrence of the class in the image. The histogram can then be seen as

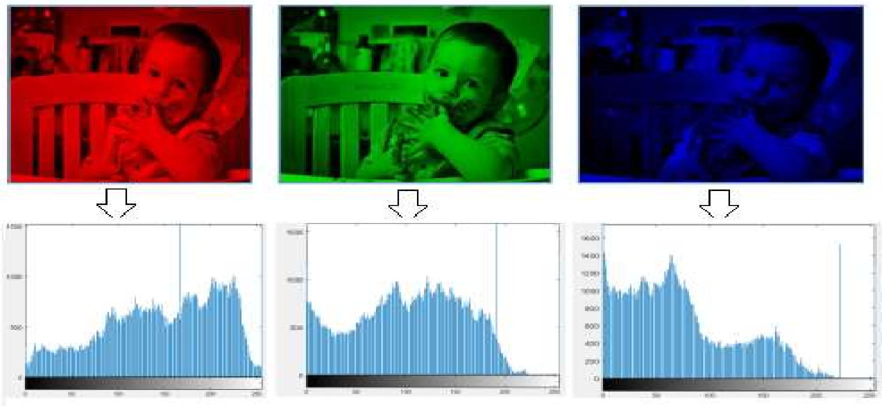


Fig. 4. Decomposition of a color image into its three channels and their corresponding Histograms.

a probability density. For color images, we can consider the histograms of the 3 components independently, but this is generally not efficient.

Rather, we construct a histogram directly in the color space. Histogram classes now correspond to a color, rather than intensity. This is called a color histogram.

4.1.2 Color Image Spatiogram

The lack of spatial information, in standard color histograms, has led us to the use of Spatiogram(s). Spatiogram(s) are generalized histograms describing more than the occurrence of the pixels in each color box, the average, and the covariance of the coordinates of the pixels, which makes it possible to capture the spatial distributions of the different image colors.

5 Tests and Results

Some experimental results are presented to evaluate our approach based on Histogram(s) and Spatiogram(s) to recognize objects in an image.

5.1 Image Database

For the database we used images from ImageNet and Pascal VOC and also, we have collected images from the Internet. All these images are divided according to their categories into test images and others into images forming the learning base. We tried to collect a large number of images according to their category so that each one can cover the maximum of images in different positions and different contrasts. We have used in our approach to do the recognition part the Support vector machines (SVM) which are a set of supervised learning techniques designed to solve discrimination and regression problems. SVMs are a generalization of linear classifiers.

Table 1. Comparison between the test image and some images from database with circular form.

Image	Histogram Similarity	Spatio Similarity
 تفاحة	**0.94**	**0.92**
 تفاحة	**0.97**	**0.93**
 تفاحة	0.22	0.15
 كرة	0.05	0.03
 كرة	0.02	0.01
 كرة	0.03	0.02
 كرة	0.06	0.05

5.2 Experimental Results

From the image database collected. We will test our algorithms on some categories of objects. The objects in the learning database are tagged to use the tags after. Subsequently, we will associate the object to recognize its histogram and its spatiogram to compare them later to those of the objects of the learning base by calculating a similarity factor. The table below shows the comparison results with different objects. From the results, we find that the values close to 1 are those associated with the images which are the most similar to the test image, despite having several objects of the same form which is circular.

5.3 Discussion

This approach has a part that serves to preserve the color information and that is consists on the use of histograms and spatiograms to help the recognition and this by a comparison between the histograms and the spatiograms with those of the unknown object. The results obtained show that the comparison between the approaches gave a precision rate of 96%. Compared with another works, [25] where the precision rate was

93% and also with [26], where the precision rate for histograms was 95% and spatiograms was also 95%, we have obtained good results that allow us to use it to generate Arabic sentences from the recognized objects.

6 Conclusion

We have presented an approach for the object recognition which is based on the color distribution using histograms and spatiograms for images in the RGB space. We can conclude that we have important results, but we will work in the future to ameliorate the results by introducing other features. The main goal of our works is to generate arabic description for an image using the recognized objects in the image.

References

1. Wallis, G., Rous, E.: Invariant face and object recognition in the visual system. Progress in Neurobiology, vol. 51, pp. 167–194 (1997)
2. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychological Review, vol. 94, 115–147 (1987)
3. Marr, D., Nishihara, H. K.: Representation and recognition of the spatial organization of three-dimensional shapes. Proceedings of the Royal Society B: Biological Sciences. vol. 200, 269–294 (1978)
4. Edelman, S.: Representation and Recognition in Vision. MIT Press, Cambridge, MA. (1999)
5. Tait, M., Williams, P., Hayward, W.: Three-dimensional object recognition is viewpoint-dependent. Nature Neuroscience, vol. 1, pp. 275–277 (1998)
6. Harris, C., Stephens, M.: A combined corner and edge detector. In: Alvey Vision Conference, Manchester - United Kingdom, pp. 147–151 (1988)
7. Klaus, A., Sormann, M., Karner, K.: Segment-Based Stereo Matching Using Belief Propagation and a Self-Adapting Dissimilarity Measure. In: Dans International Conference on Pattern Recognition, Graz, Autriche, août, vol. 3, pp. 15–18, (2006)
8. Scharstein, D., Szeliski, R.: A Taxomomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. International Journal of Computer Vision, vol. 47, no. 1, pp. 7–42 (2002)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016)
10. Chabot, F., Chaouch, M., Rabarisoa, J., Teulière, C., Chateau, T.: Deep MANTA: A Coarse-to-fine Many-Task Network for joint 2D and 3D vehicle analysis from monocular image (2017)
11. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun Y.: Integrated recognition, localization and detection using convolutional networks. In: International Conference on Learning Representations (2014)
12. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.: Single shot multibox detector. In: European Conference on Computer Vision (2016)

13. Redmon, J., Farhadi, A.: YOLO9000: Better, faster, stronger. In: Computer Vision and Pattern Recognition, pp. 7263–7271 (2017)
14. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition (2001)
15. Dollar, P., Tu, Z., Perona, P., Belongie, S.: Integral channel features (2009)
16. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition (2005)
17. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: Neural Information Processing Systems (2012)
18. Uijlings, J. R., van de Sande, K. E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. International Journal of Computer Vision (2013)
19. He, K., Gkioxari, G., Dollar, P., Girshick, R.: Mask R- CNN. In: International Conference on Computer Vision, pp. 2961–2969 (2017)
20. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: European Conference on Computer Vision (2014)
21. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Neural Information Processing Systems (2015)
22. Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Computer Vision and Pattern Recognition vol. 37, no. 9, pp. 1904–1916 (2017)
23. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: Computer Vision and Pattern Recognition (2016)
24. Shrivastava, A., Sukthankar, R., Malik, J., Gupta, A.: Beyond skip connections: Top-down modulation for object detection (2016)
25. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3128–3137 (2015)
26. Lin, T. Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection (2017)
27. Vaillant, R., Monroca, C., Le-Cun, Y.: Original approach for the localisation of objects in images. IEEE Proceedings on Vision, Image, and Signal Processing, vol. 141, no. 4, pp. 245–250 (1994)
28. Farhani, N., Naim, T., Zrigui, M.: Image to text conversion: state of the art and extended work. 2017 In: IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), pp. 937–943, (2017)
29. Terbeh, N., Zrigui, M.: Vocal pathologies detection and mispronounced phonemes identification: Case of Arabic continuous speech. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pp. 2108–2113 (2016)
30. Ayadi, R., Maraoui, M., Zrigui, M.: A Survey of Arabic Text Representation and Classification Methods. Research in Computing Science. vol. 117, pp. 51–62 (2016)
31. Terbeh, N., Achraf, M., Ben, M., Zrigui, M.: Probabilistic approach to Arabic speech correction for peoples with language disabilities. International Journal of Information Retrieval Research (IJIRR) vol. 5, no. 4, pp. 1–18 (2015)

32. Mansouri, S., Charhad, M., Zrigui, M.: Arabic text detection in news video based on line segment detector. Research in Computing Science. vol. 132, pp. 97–106 (2017)

ISSN 1870-4069

Machine Translation Evaluation Metrics for Recognizing Textual Entailment

Tanik Saikh[1], Asif Ekbal[1], Debajyoty Banik[2],
Pushpak Bhattacharyya[3]

[1]Indian Institute of Technology, Patna,
India

[2]Kalinga Institute of Industrial Technology, Bhubaneswar,
India

[3]Indian Institute of Technology Bombay
India

debajyoty.banik@gmail.com, pb@cse.iitb.ac.in
{tanik.srf17, asif}@iitp.ac.in

Abstract. In this paper we propose a method for Recognizing Textual Entailment (RTE) that makes use of different machine translation (MT) evaluation metrics, namely *BLEU, METEOR, TER, WER, LE-BLEU, NIST* and *RIBES* and different versions of summary evaluation metric ROUGE, namely *ROUGE-N, ROUGE-S, ROUGE-W, ROUGE-L* and *ROUGE-SU* in a machine learning framework. Our main motivation of this paper is to investigate how MT evaluation metrics (which is generally used to judge the quality of an MT output), summary evaluation metrics (which is generally used to measure the quality of system generated summary) can be effective for determining TE relation between a pair of text snippets. Experiments on the datasets released as part of the shared task for recognizing textual entailment, RTE-1, RTE-2, RTE-3, RTE-4 and RTE-5 show the encouraging performance. We also performed a deeper comparative analysis of relevance of MT and summary evaluation metrics for the task of Textual Entailment (TE).

Keywords: Evaluation metrics, textual binding, automatic translation.

1 Introduction

One of the utmost challenging problems in the filed of Natural Language Processing (NLP) is to deal with language variability, that means there can be multiple ways to express a simple matter. Over the years researchers have been investigating a common framework which will be able to capture such language variability. Textual Entailment (TE) is an effective way to capture such language variability. Textual entailment requires complex linguistic analysis. TE was first introduced by [9] in the first track of recognizing textual entailment organized by *National Institute of Standard and Technology (NIST)*.

In this track TE was first defined as follows: suppose there are two texts fragments expressed as *Text (T)* and *Hypothesis (H)*. It is said that: ***T** entails **H** if, typically, a human reading **T** would infer that of **H** is most likely to be true*. For example, the text *T = "Mahatma Gandhi's assassin happened"* entails the hypothesis *H = "Mahatma Gandhi was died"*; obviously, if there exists one's assassin, then this person is died. Similarly, *T = "Mary lives in Germany"* entails *H = "Mary lives in Europe"*. On the other hand, *T = "Mary lives in Europe"* does not entail *H = "Mary lives in India"*.

There were many international conferences and evaluation tracks have been organized such as at Pattern Analysis, Statistical Modeling and Computational Learning (PASCAL)[4], Text Analysis Conferences (TAC)[5] organized by the United States National Institute of Standards and Technology (NIST), Evaluation Exercises on Semantic Evaluation (SemEval)[6], National Institute of Informatics Test Collection for Information Retrieval System (NTCIR) [7] since from the year of 2005.

These conferences and workshops produced many research articles which cover many approaches, varying from Lexical [9] [23], Syntactic [34] and semantics [4]. There are many applications in the field of NLP where TE can be employed, e.g. Machine Translation (MT) [16], Question-Answering (QA) [12] and Summarization [27] and many more. In MT evaluation, the machine generated output should entailed with the reference one.

In Question-Answering (QA), the answer produced by a machine must entail with that particular question. In summarization, the machine produced summary should entail with the reference one. Building an MT evaluation metric with the help of TE is a vital task. The proposed study makes use of various MT evaluation metrics and automatic summary evaluation metrics as features in machine learning framework.

1.1 Motivation

The MT evaluation metrics and automatic summary evaluation metrics are meant to investigate the quality of translation and summarization. It essentially does that by measuring similarity between the two (machine produced outputs and references) comparing piece of text fragments. We use these metrics to determine TE relationship. The study in [31] made use of very well established similarity metrics like Cosine, Jaccard, Dice, Overlap etc. and two MT evaluation metrics, namely BLEU [24] and METEOR [20, 1] for TE using RTE-1, RTE-2 and RTE-3 datasets.

This shows that MT evaluation metrics performed at per the ordinary similarity metrics in predicting the TE relation. Another study [30] made use of the same kind of similarity metrics along with the MT evaluation metrics on Indian languages (namely Hindi, Punjabi, Telugu and Malayalam) datasets to detect the paraphrase relation between a pair of sentences. The evaluation was performed on the datasets of the shared task *Detecting Paraphrases in Indian Languages (DPIL)* of *Forum for Information Retrieval Evaluation (FIRE-2016)*.

[4] http://pascallin.ecs.soton.ac.uk/Challenges/

[5] http://www.nist.gov/tac/tracks/index.html

[6] http://semeval2.fbk.eu/semeval2.php

[7] http://research.nii.ac.jp/ntcir/ntcir-9/

The study of [29] proposed a model which would be able to predict the TE relation between the two sentences (in RTE-1, RTE-2, RTE-3, RTE-4 and RTE-5 datasets) based on the MT evaluation metrics (BLEU, METEOR and TER) and one summary evaluation metric (ROUGE). Hence if BLEU, TER, and METEOR can take part in predicting TE relation between a pair of text snippets, there are other metrics too which could take part in deciding entailment relation between two piece of texts. The set of features applied here is very new to predict the TE relation in machine learning framework. There are also some works on *Microsoft Research Paraphrase (MSRP)* Corpus by [22] to detect paraphrases using MT evaluation metrics. So people are working in this line.

1.2 Related Work

There are umpteen number of research works carried out that could be found in literature on the datasets released in the shared task for recognizing textual entailment i.e. RTE-1, RTE-2, RTR-3, RTE-4 and RTE-5. Literature shows, [26] obtained the best accuracy in RTE-1 by the methods of word overlapping. They made use of BLEU, whose scores were used to assign “yes” or “no” class entailment decision based on some thresholds. The thresholds were learned based on some heuristics which were devised using the datasets. They obtained an accuracy of 70% on the dataset. The method defined in [15] achieved the best accuracy of 75% on RTE-2 dataset. The study proposed by the system made use of lexical and syntactic matching.

In RTE-3 dataset the best result was obtained in [14] using lexical alignment, knowledge extraction method, discourse commitment etc. The system proposed by [17] obtained the highest accuracy of 68.5%. The key concept of the participant’s system is to map each word of hypothesis with one or more words of the text. They extensively made use of knowledge bases, namely DIRT, WordNet, VerbOcean, Wikipedia and Acronyms database etc. The system defined in [18] achieved the highest accuracy of 68.33% in RTE-5. The approach defined here is same as the approach defined in [17]. Additionally, they processed the LingPipe output with GATE in order to identify some of the named entity categories (e.g. nationality, language, job) which are within the scope of LingPipe.

Apart from the above cited works on TE and paraphrase detection there are few more works found in literature. The task described in [11] made use of BLEU, NIST, TER and Position independent word error rate (PER) to build a classifier which will be able to predict paraphrase relation between a pair of texts and also the entailment relation. They made use of *MSRP Corpus* and *RTE-1* for detection of paraphrase and entailment respectively. The work of [22] made use of *Microsoft Paraphrase Detection Corpus (MSRP)* and *Plagiarism Detection Corpus (PAN)* to re-examine the idea that automatic metrics which are generally used for judging the quality of a translation can also perform for the task of paraphrase detection. They used BLEU, NIST, TER, TERp [33], METEOR, SEPIA [13], BADGER [25], MAXSIM [5] metrics.

2 Feature Analysis

Features play a pivotal role in any machine learning assisted experiment. Hence identifying right combination of features which yield the best accuracy is the vital task. The following subsections define the features employed in the proposed study.

2.1 MT Evaluation Metrics

MT evaluation metrics generally used to judge the closeness between the machine translated output and the gold standard reference one. The more the closeness between them, the better is the translation system output.

Over the years, MT community proposed various metrics, namely BLEU [24], METEOR [20, 1], NIST [10], TER [32], Word Error Rate (WER) [37], Position independent word Error Rate (PER) and General Text Matcher (GTM) etc.We have incorporated almost all the metrics available in this study. We describe each of them in the following points:

1. BLEU: Bilingual Evaluation Understudy(BLEU) [24] is a metric which is perhaps the most popular MT evaluation metric developed by IBM. It measure the similarity between MT output and reference sentences by computing the n-gram precision between those sentences. Mathematically, it can be expressed as follows:

$$BLEU = BP \cdot \exp\left(\sum_{N}^{n=1} w_n log_{p_n}\right). \tag{1}$$

where, w_n is positive weights, summing to one; p_n is modified n-gram precisions. BP is brevity penalty computed as:

$$BP = \begin{cases} 1 & \text{if } c > r, \\ e^{\left(1-\frac{r}{c}\right)} & \text{if } c \leq r. \end{cases} \tag{2}$$

where, candidate translation length is c; and length of the effective reference corpus is r. Generally BLEU measures the similarity between the two sentences by computing n-gram matching. It can be termed as precision oriented metric. Similarity can be a way to detect TE relation, as it produce similarity, we wield this as a feature in this study.

2. LE-BLEU: *LE-BLEU* is also an MT evaluation metric which is based on n-gram matching, considered suitable for highly compounding languages [36]. It is an extension of BLEU, which consider fuzzy matches between two n-grams. It is supposed to better correlate with human judgment. The main drawback of BLEU is in exact n-gram matching. It could be possible not to match exact n-gram but having similar meaning. Such case LE-BLEU perform well than BLEU. LE-BLEU calculation is a kind of fuzzy calculation, whereas the calculation of BLEU is crisp kind of technique. So, there is great chance to get BLEU score zero, (if it is considering 4-gram matching, but machine up to 3-gram) where LE-BLEU provides satisfactory score. Thus, LE-BLEU can be a smart feature to recognize textual entailment between a piece of texts.

3. RIBES: Rank-based Intuitive Bilingual Evaluation Score (RIBES) [19] is also very useful for judging the MT output. Its calculation is based on significantly penalized word order mistakes and rank correlation coefficients. It is very effective for evaluating the accuracy score between distant sentence pairs. So, the similarity score between two distant sentences can easily predict their entailment.

4. NIST: The name NIST [10] came from *US National Institute of Standard and Technology*, which is used to evaluate between different text pairs. BLEU considers each n-gram to be of equal weight whereas NIST considers only the informative n-grams. NIST also differs from BLUE in its calculation of the brevity penalty in so far as small variations in translation length do not impact the overall score as much. As it also measures the similarity between the pair of text snippets, we exert this as feature.

5. METEOR: METEOR (Metric for Evaluation of Translation with Explicit Ordering) [20, 1] is a metric which measures the similarity between two sentences by computing the maximum-cardinality matching between those sentences. This match is used to compute the coherent based penalty. This computation is done by assessing the extent to which the matched words between texts constitute well ordered coherent “chunks”. It considers lexical matching and synonym matching from the WordNet. As it takes both the approaches to measure the similarity into account, it could be an interesting feature in our study.

6. TER: Translation Error Rate (TER) [32] is also a metric to evaluate the performance of an MT output which is introduced in ”Global Autonomous Language Exploration (GALE) Program” MT task. The central concept behind this metric is that edits required to change a hypothesis translation into reference translation. It generally produces the error rate by measuring the edit operations required to transfer the MT output to reference translation. Hence we get the similarity by taking complement of the error rate as shown in the following equation 3:

$$TER = \frac{\text{Number of edits}}{\text{average number of reference words}}. \tag{3}$$

Correctness is important to detect textual entailment, but it is not sufficient for this task. Error between two texts also important for recognizing textual entailment. So, we have considered TER and WER to fulfill its requirement.

7. WER: Word error rate (WER) [37] is also another very popular MT metric, which is also used in speech recognition. The metric works on word level and it is based on Levenshtein distance. It’s computation is based on the minimum substitutions, deletions and insertions that have to be performed to convert the generated text into the reference text. It can be computed by the following formula 4: where *S:# of substitutions, D:# of deletions, I:# of of insertions, N:# of words in the reference*:

$$WER = \frac{S + D + I}{N}. \tag{4}$$

2.2 Summary Evaluation Metrics

Summary evaluation metrics are generally used to judge the quality of machine generated summary of a document which is generated automatically following an algorithm. *Recall-Oriented Understudy for Gisting Evaluation (ROUGE)* [21] is one of the most popular and widely accepted metric to evaluate summary. It comes up with it's several versions, we utilize all those versions here. Important ingredient for scoring for this metric is overlapping units. Units refer to n-gram word sequence word pairs between the hypothesis and reference sentences.

1. ROUGE-N: It computes the similarity score by measuring n-gram matching. Specifically, it is n-gram recall between a set of reference and candidate document. The mathematical equation of this metric is as follows:

$$ROUGE - N = \frac{\sum_{S \in \{\text{reference documents}\}} \sum_{\text{gram} \in}^{S} \text{Count}_{\text{match}}\big(\text{gram}\big)}{\sum_{S \in \{\text{reference documents}\}} \sum_{\text{gram} \in S} \text{count(gram)}}. \tag{5}$$

where, count match(gram) is the maximum number of n-grams co-occurring in a candidate and a set of reference documents. The intuition behind using this metric is that it corresponds to the recall version of BLEU. So, we take into account precision from BLEU and recall from this metric.

2. ROUGE-L: It is basically the Longest Common Subsequence (LCS) based statistics. LCS is used as approximate string matching algorithm. To compare similarity between the hypothesis and reference documents normalized pairwise LCS is used in [28]. The higher the common matching between two texts the more the chances of the two texts to be textually entailed.

3. ROUGE-W: It is weighted LCS that favors consecutive LCSs. Problem with basic LCS is that it is unable to differentiate LCSs of different spatial relations. As it calculates similarity by taking weighted LCS into account, it can be an effective feature to predict entailment.

4. ROUGE-S: ROUGE-S is based on skip-bigram co-occurrence. It calculates the similarity score between two piece of texts, by considering bi-gram matching irrelevant of word order. We use this as a feature in our model.

5. ROUGE-SU: ROUGE-SU is the combination of skip-gram and unigram. It computes similarity by taking both of these into consideration. It is a different version to measure the similarity between a pair of texts snippets. This is also used as a feature.

3 Experimental Setup and Results

In this section we present preprocessing module, description of the datasets, experimental procedure, results, discussions and comparisons with the state-of-the arts.

Table 1. Statistics of the Datasets.

	# of T-H pair	
	Development	**Test**
RTE-1	567	800
RTE-2	800	800
RTE-3	800	800
RTE-4	0	1000
RTE-5	600	600

3.1 Preprocessing Module

Data are full of noisy by default. This performs the cleaning operation of such noise from the T-H pair contained in the datasets. We also removed the white spaces (if any) from the datasets. The example below shown is a T-H pair in the development set (taken from RTE-1).

```
<pair id="78" value="FALSE" task="IR"> <t>Clinton&apos;s
new book is not big seller here.</t>
<h>Clinton&apos;s book is a big seller.</h>
</pair>
```

Here `&apos;s` are replaced by “’” in the sentences, and then further it was converted into it’s expanded form i.e. `Clinton&apos;s` converted into Clinton’s.

3.2 Dataset

We use the datasets released in the shared task for recognizing textual entailment i.e. RTE-1, RTE-2, RTE-3, RTE-4 and RTE-5. The datasets of RTE-1, RTE-2 and RTE-3 correspond to the binary-class classification problem, whereas the datasets of RTE-4 and RTE-5 denote the ternary class classification problem. In our work we consider both binary and ternary classification. In Table 1 we show the number of T-H pairs present in the datasets.

3.3 Experiments

We extract T-H pairs from the datasets of RTE-1, RTE-2, RTE-3, RTE-4 and RTE-5. we calculate the similarity between T and H by exploiting the set of features that we already discussed. The scores obtained from each metric for a particular T-H pair are considered as feature value which are subjected for classifier’s training and/or testing. As base learning algorithms, we use Support Vector Machine (SVM) [35, 6], Multilayer Perceptron (MLP) [2, 8], Logistic Regression [7] and Random forest (RF) [3]. We use the classifiers as available in weka[8]. The models are used to predict a class for an unknown T-H pair. We report the evaluation figures on the test set. The system predicts a class to each instances (T-H pair) of the test set. For RTE-4 we perform 10-fold cross validation results as we don’t have access to the test dataset.

[8] http://www.cs.waikato.ac.nz/ml/weka/

Table 2. Results on test set of different datasets.

	Our System Accuracies(%)				Best Results(%)(Participant)
	SVM	Logistics	MLP	RF	
RTE-1	55.5	53.87	52.75	56.37	70 [26]
RTE-2	60.12	59.5	57	59.12	75 [15]
RTE-3	61.87	62.25	60.5	60.37	80 [14]
RTE-4	54.4	54.5	51.4	52.3	68.5 [17]
RTE-5	50	34.83	52.2	46.33	68.33 [18]

Table 3. Comparison results with baseline system.

Datasets	Baseline [29](%)	Proposed approach(%)
RTE-1	54	55.5
RTE-2	55	60.12
RTE-3	60	62.25
RTE-4	52	54.4
RTE-5	51	52.2

3.4 Results, Discussions and Comparisons

We evaluate all the three models on all the three datasets. We calculate *True Positive (TP), False Positive (FP), False Negative (FN)* and *True Negative (TN)*. We also report the accuracy of the system. We depict the accuracies obtained by the proposed system and the state-of-the-art models on different datasets in Table 2. In RTE-1 the best accuracy of our system is 56.37% using Random Forest (RF) compared to the best result reported as 70% by [26]. For RTE-2 we get an accuracy of 60.12% with SVM, whereas the best accuracy reported is 75% by [15]. For RTE-3, we obtain the highest accuracy of 62.25% in Logistics Regression framework, however the best accuracy reported in this track is 80% by [14]. For RTE-4, we obtain an accuracy of 54.5% using Logistic Regression classifier compared to the best reported value of 68.5% by [17].

The system with MLP yields an accuracy of 52.2% in RTE-5, whereas an accuracy of 68.33% was reported as the best result by [18]. It is to be noted that however we obtain relatively less accuracies compared to the state-of-the-arts, novelty of our proposed techniques is in the use of different MT and summary evaluation metrics for classifier's training for TE. Table 2 shows the evaluation results of different classifiers in our proposed system. The last column shows the state-of-the-art results obtained by the different participating systems in the respective tracks. For RTE-1 it is observed that Random Forest produces the best result among all the classifiers. For RTE-2, SVM model attains the best accuracy.

In RTE-3 and RTE-4, logistic regression model yield the highest performance. In RTE-5 MLP produces the best result. It is to be noted that different classifiers produces the best result for the different datasets. Comparisons to the baseline [29] models are presented in table 3. The system reported in [29] made use of all the datasets as what our proposed system exploits. The proposed system makes use of the baseline features as well as the others extracted from NIST, LE-BLEU, WER, RIBES, ROUGE-N, ROUGE-S, ROUGE-L, ROUGE-W and ROUGE-SU.

Table 4. Comparison between MT and Summary Evaluation metrics.

		Accuracy (%)	
Dataset	**Classifier**	**MT**	**Summary**
RTE-1	RF	56.25	50.37
RTE-2	SVM	58.37	60.5
RTE-3	Logistic	62.12	59.5
RTE-4	Logistic	55.1	51.5
RTE-5	MLP	53	50

Table 5. Features ablation study.

	BLEU	LE-BLEU	RIBES	NIST	WER	TER	METEOR	ROUGE-N	ROUGE-L	ROUGE-W	ROUGE-S	ROUGE-SU
RTE-1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
RTE-2	X	✓	X	X	X	X	N	N	N	X	✓	✓
RTE-3	N	X	X	X	✓	✓	N	X	N	X	X	X
RTE-4	N	✓	X	✓	X	N	✓	N	N	✓	✓	N
RTE-5	X	X	X	X	X	X	✓	✓	N	✓	X	N

This shows that our proposed approach is more effective than our baseline model which infers that usages of MT and summary evaluation metrics for the proposed task are, indeed, effective. We also performed a deeper analysis by comparing the result obtained by taking MT metrics alone as feature and the result obtained by taking summary evaluation metrics alone as feature. We run the model by the best performing classifier in each dataset and obtained and obtain two sets of results for MT and summary evaluation metric each. The results are shown in the following table:

3.5 Features Sensitivity Analysis

Features are very precious in any machine learning assisted experiment. In order to embellish the contribution of each feature to our predicting classes, we perform Ablation Study of features where we switch off a feature and then evaluate the model with the rest set of features and compare with the accuracy obtained by whole set of features (including the particular feature). Table 5 represents datasets in columns and row heading represent the features.

In this Table the ”✓” represents that the corresponding feature has a positive effect on the performance, whereas ”N” indicates that the particular features seem to have no effect and ”X” denotes, the features seem to have negative effect on the performance. The table shows that, for RTE-1, all the features seem to have positive contributions. In RTE-2 *LE-BLEU*, *ROUGE-S* and *ROUGE-SU* features seem to be the most contributing features. The features, namely *METEOR*, *ROUGE-N* and *ROUGE-L* do not contribute significantly.

On the other hand, the features, namely *RIBES*, *NIST*, *WER*, *TER* and *ROUGE-W* seem to have negative effect on the performance. In RTE-3 datasets, only *WER and TER* are found to contribute more, *BLEU, METEOR and ROUGE-L* are found to be neutral, and *LE-BLEU, RIBES, NIST, ROUGE-N, ROUGE-W and ROUGE-S and ROUGE-SU* are found to be the features with negative effect.

Table 6. Results with contributing features only.

Datasets	Accuracies(%)
RTE-1	55.5
RTE-2	58.8
RTE-3	57.62
RTE-4	55.3
RTE-5	46.16

Table 7. Syntactic Parsing of T and H.

Syntactic Parsing	
T: John loves Merry.	H: Merry loves
(ROOT	(ROOT
(S	(S
(NP (NNP John))	(NP (NNP Merry))
(VP (VBZ loves)	(VP (VBZ Loves)
(NP (NNP Merry)))	(NP (NNP John)))
(. .)))	(. .)))

For RTE-4, *LE-BLEU, NIST, METEOR, ROUGE-W and ROUGE-S* are found to be the most effective features. For RTE-5, *METEOR, ROUGE-N and ROUGE-W* features contribute most. It is to be noted that, *LE-BLEU, METEOR, ROUGE-W and Rouge-S* are the features which are found to be the contributing features in most of the datasets. We perform another set of experiments by training and/or testing the classifier by considering only the contributing features. Results of these models are reported in Table 6. Please note that we report only the results of the best performing classifier (for each dataset).

3.6 Error Analysis

We perform error analysis to understand the shortcomings of our proposed method. Our system makes use of MT and summary evaluation metrics in a machine learning framework. Most of these metrics are based on lexical matching that may not be sufficient to capture the textual similarity always. These are not able to capture the syntactic and semantic ambiguities present in the corpus. Lexical matching ache from a drawback, sometimes it produces very high score for non-textually entailed text pair.

For the following example, *T: John loves Merry* and *H: Merry loves John*, n-gram matching produces a very high score and consequently, the system will mark that pair as entailed. Unigram and bigram matchings between T and H produce 3/3 = 1 and 0/3 = 0 scores, respectively. According to unigram matching the sentence pair is textually entailed, however they are actully should not be. On the other hand, if we parse T and H using a Stanford parser [9], it will produce the parsing information of that particular T and H as shown in the table 7.

[9] http://nlp.stanford.edu:8080/parser/index.jsp

Here none of the left or right child matches, hence they (T and H) are considered to be not textually entailed. Hence, syntactic information plays a vital role in determining TE relations. The system needs to be updated in this front. Let us consider the another example taken from the RTE-3 development set.

```
   <pair id="251" entailment="YES" task="IR" length="short">
<t>Estimates vary widely, but it is believed there are up to
100 million children toiling in homes, factories, shops, fields,
brothels and on the streets of rural and urban India.</t>
<h>Child labor is widely used in Asia.</h> </pair>
```

There is only one common token (i.e **widely**) between T and H in the above example, so if we consider for lexical matching between those pair, the system will produce a very low score which is not sufficient to tag them (T-H pair) as textually entailed. However this pair should be defined as textually entailed. This needs further investigation.

4 Conclusion and Future Work

In this paper, we have proposed a system for recognizing textual entailment between a pair of text expressions which exploits MT evaluation metrics *namely BLEU, NIST, RIBES, LE-BLEU, TER, WER and METEOR* and summary evaluation metrics (*namely ROUGE-N, ROUGE-S, ROUGE-L, ROUGE-W and ROUGE-SU*) as features in a supervised machine learning framework. We develop models based on SVM, Logistic Regression, MLP and Random Forest.

Experiments performed on different benchmark datasets of RTE-1, RTE-2, RTE-3, RTE-4 and RTE-5 tracks show that our proposed approach attain encouraging performance. We believe that the proposed system is novel as the current study makes use of the features which are new of it's kind. The proposed system has great potential of applications for evaluating the MT and summary outputs.

The proposed study is in the opposite direction, which try to correlate MT and summary evaluation metrics with TE. The experiments reveal that MT and summary evaluation metrics which are generally use to judged the quality of machine produced translation and summary respectively, have a strong correlation which can also effectively take part in taking entailment decision between a pair of text snippets. Future works are directed towards the following dimensions:

- Will incorporate more such metrics in the existing system to build a more robust TE system.
- Will incorporate deep learning concepts in the existing system and want to make a comparative study.
- Planning to incorporate these MT and summary evaluation metrics in semantic textual similarity, which is another interesting problem to study.
- Planning to the work in reverse direction i.e. to build an MT evaluation metric by exploiting a robust TE system which will be based on deep learning approach.

References

1. Banerjee, S., Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization at the 43rd Annual Meeting of the Association of Computational Linguistics (ACL-2005). pp. 65–72 (2005)
2. Becerra, R., Joya, G., García Bermúdez, R. V., Luis, V., Rodríguez, R., Pino, C.: Saccadic points classification using multilayer perceptron and random forest classifiers in EOG recordings of patients with ataxia SCA2. Advances in Computational Intelligence. Lecture Notes in Computer Science , vol. 7903, no. 3 (2013)
3. Breiman, L.: Random forests. Machine Learning , vol. 45, no. 1, pp. 5–32 (2001)
4. Burchardt, A., Reiter, N., Thater, S., Frank, A.: A semantic approach to textual entailment: System evaluation and task analysis. In: Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. pp. 10–15. Association for Computational Linguistics (2007)
5. Chan, Y., Ng, H.: Maxsim: A maximum similarity metric for machine translation evaluation. Proceedings of ACL-HLT. pp. 55–62 (2008)
6. Chang, C. C., Lin, C. J.: Libsvm: A library for support vector machines. Association for Computing Machinery. Intelligence, Systems, Technology and Management , vol. 2, no. 3, pp. 1–27 (2011)
7. Collins, M., Schapire, R. E., Singer, Y.: Logistic regression, adaboost and bregman distances. Machine Learning , vol. 48, no. 1–3, pp. 253–285 (2002)
8. Costa, W., Garcia Fonseca, L. M., Sehn Körting, T.: Classifying grasslands and cultivated pastures in the brazilian cerrado using support vector machines, multilayer perceptrons and autoencoders. In: Machine Learning and Data Mining in Pattern Recognition - 11th International Conference. pp. 187–198 (2015)
9. Dagan, I., Glickman, O., Magnini, B.: The pascal recognising textual entailment challenge. In: Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment. pp. 177–190. Springer-Verlag (2006)
10. Doddington, G.: Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In: Proceedings of the Second International Conference on Human Language Technology Research. pp. 138–145. Morgan Kaufmann Publishers Inc. (2002)
11. Finch, A., Sook Hwang, Y., Sumita, E.: Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In: Proceedings of the Third International Workshop on Paraphrasing. pp. 17–24 (2005)
12. Green, B. F., Wolf, A. K., Chomsky, C., Laughery, K.: Baseball: An automatic question-answerer. In: Papers Presented at the May 9-11, 1961, Western Joint IRE-AIEE-ACM Computer Conference. pp. 219–224. Association for Computing Machinery (1961)
13. Habash, N., Elkholy, A.: SEPIA: Surface span extension to syntactic dependency precisionbased MT evaluation. Proceedings of the NIST metrics for machine translation workshop at the association for machine translation in the Americas conference (2008)
14. Hickl, A., Bensley, J.: A discourse commitment-based framework for recognizing textual entailment. In: Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. RTE ’07, Stroudsburg, PA, USA, Association for Computational Linguistics (2007). pp. 171—-176 (2007)
15. Hickl, A., Bensley, J., Williams, J., Roberts, K., Rink, B., Shi, Y.: Recognizing textual entailment with lccs groundhog system. In: Proceeding of the Second PASCAL Challenges Workshop. (2005)

16. Hutchins, W. J., Somers, H. L.: An introduction to machine translation. London: Academic Press (1992)
17. Iftene, A.: UAIC Participation at RTE4. Text Analysis Conference Workshop. National Institute of Standards and Technology pp. 17–19 (2008)
18. Iftene, A., Moruz, M.: UAIC participation at RTE5. Text Analysis Conference Workshop. National Institute of Standards and Technology (2009)
19. Isozaki, H., Hirao, T., Duh, K., Sudoh, K., Tsukada, H.: Automatic evaluation of translation quality for distant language pairs. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. pp. 944–952. Association for Computational Linguistics (2010)
20. Lavie, A., Agarwal, A.: METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In: Proceedings of the Second Workshop on Statistical Machine Translation. pp. 228–231. Association for Computational Linguistics (2007)
21. Lin, C. Y.: ROUGE: A package for automatic evaluation of summaries. In: Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004). pp. 74–81 (2004)
22. Madnani, N., Tetreault, J., Chodorow, M.: Re-examining machine translation metrics for paraphrase identification. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 182–190 (2012)
23. Pakray, P., Bandyopadhyay, S., Gelbukh, A.: Lexical based two-way RTE system at RTE-5. In: System Report, TAC RTE Notebook. (2009)
24. Papineni, K., Roukos, S., Ward, T., Zhu, W.-J.: BLEU: A Method for Automatic Evaluation of Machine Translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. pp. 311–318. Association for Computational Linguistics (2002)
25. Parker., S.: A New Machine Translation Metric. Proceedings of the Workshop on Metrics for Machine Translation at AMTA. (2008)
26. Perez, D., Alfonseca, E.: Application of the bleu algorithm for recognising textual entailments. In: In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment. (2005)
27. Radev, D. R., McKeown, K. R.: Generating natural language summaries from multiple on-line sources. Computational Linguistics , vol. 24, no. 3, pp. 469–500 (1998)
28. Saggion, H., Radev, D., Teufel, S., Lam, W.: Meta-Evaluation of Summarization in a Cross–Lingual Environment Using–Based Metrics. Proceedings of COLING-2002, Taipei, Taiwan (2002)
29. Saikh, T., Naskar, S., Ekbal, A., Bandyopadhyay, S.: Textual Entailment using Machine Translation Evaluation Metrics. In: Computational Linguistics and Intelligent Text Processing - 18th International Conference on Computational Linguistics and Intelligent Text Processing, CICLing 2017 (2017)
30. Saikh, T., Naskar, S. K., Bandyopadhyay, S.: JU_NLP@DPIL-FIRE 2016: Paraphrase Detection in Indian Languages - A machine Learning Approach. In: Working notes of FIRE 2016 - Forum for Information Retrieval Evaluation, Kolkata, India. pp. 275–278 (2016)
31. Saikh, T., Naskar, S. K., Giri, C., Bandyopadhyay, S.: Textual entailment using different similarity metrics. In: Computational Linguistics and Intelligent Text Processing - 16th International Conference, CICLing 2015, Cairo, Egypt, April 14-20, Proceedings, Part I. pp. 491–501 (2015)
32. Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J.: A Study of Translation Edit Rate with Targeted Human Annotation. In: In Proceedings of Association for Machine Translation in the Americas. pp. 223–231 (2006)
33. Snover, M., Madnani, N., Dorr, B., Schwartz, R.: TER-plus: Paraphrase, semantic, and alignment enhancements to translation edit rate. Machine Translation. , vol. 23, pp. 117—-127 (2009)

34. Vanderwende L., D. W.: What Syntax Can Contribute in the Entailment Task. In: Quiñonero-Candela J., Dagan I., Magnini B., d'Alché-Buc F. (eds) Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment. , vol. 3944, pp. 205–216. Springer (2006)
35. Vapnik, V. N.: The Nature of Statistical Learning Theory. Springer-Verlag New York, Inc. (1995)
36. Virpioja, S., Grönroos, S.-A.: LeBLEU: N-gram-based Translation Evaluation Score for Morphologically Complex Languages. In: WMT@ EMNLP. pp. 411–416 (2015)
37. Wang, Y.-Y., Acero, A., Chelba, C.: Is word error rate a good indicator for spoken language understanding accuracy (2003)

ISSN 1870-4069

Classification and Annotation of Arabic Factoid Questions: Application to Medical Domain

Essia Bessaies, Slim Mesfar, Henda ben Ghezala

University of Manouba,
Tunisia

essiabessaies@gmail.com, mesfarslim@yahoo.fr,
henda.benghezala@ensi.rnu.tn

Abstract. Our question answering system is based on a linguistic approach, using NooJ's linguistic engine in order to formalize the automatic recognition rules and then apply them to a dynamic corpus composed of medical journalistic articles. We started by putting in place rules that identify and annotate the different medical entities. The module called entity recognition is able to find references to people, places and organizations, diseases, viruses, as targets to extract the correct answer from the user. These annotations are used in our system in order to identify these answers associated with the extracted named entities. The system is mainly based on a set of local grammars developed for the identification of different structures of phrases to extract the right answer. In addition, we present a method for analyzing medical questions and the approach to finding an answer to a submitted question based on the linguistic approach. The precision and recall show that the actual results are encouraging and could be integrated in a more general Arabic question answering system.

Keywords. Information extraction, medical questions, Arabic language, local grammar, named entities, journalistic articles.

1 Introduction

Nowadays, the medical domain has a high volume of electronic documents. The exploitation of this large quantity of data makes the search of specific information complex and time consuming. This complexity is especially evident when we seek a short and precise answer to a human natural language question rather than a full list of documents and web pages. In this case, the user requirement could be a Question Answering (QA) system which represents a specialized area in the field of information retrieval.

The goal of a QA system is to provide inexperienced users with flexible access to information allowing them to write a query in natural language and obtain not the documents which contain the answer, but its precise answer passage from input texts. There has been a lot of research in English as well as some European language QA systems. However, Arabic QA systems (Brini et al, .2009) could not match the pace

Table 1. Question answering system.

	Arabic language	**Linguistic**	**learning-based**	**Factoid Question**	**No-Factoid**	**Semantic analysis of the question**	**Without Semantic**
QARAB (Hammou et al. 2002)	✓		✓	✓	✓		✓
ArabiQA (Ben NAjiba et al. 2007)	✓	✓		✓		✓	
QALC (Ferret et al., 2000; Ferret et al., 2001a)		✓		✓		✓	
QUANTUM (Plamondon et al. 2002)			✓	✓	✓	✓	
AQAS (Mohammed F, et al 1993)	✓		✓	✓	✓	✓	
Qristal (Laurent et al., 2005)			✓	✓	✓	✓	
Esculape (Embarek,2009)		✓	✓	✓	✓	✓	
Piquant (Chu-Carroll et al., 2002)			✓	✓		✓	
JAVELIN (Nyberg et al., 2002)			✓	✓	✓	✓	
InsightSoft (Soubbotin et al., 2002)			✓	✓		✓	
PowerAnswer (Moldovan et al., 2002)		✓	✓	✓	✓	✓	
Webcoop(Ben amara, 2004)			✓		✓	✓	
Citron (Falco et al, 2014)			✓	✓	✓	✓	

due to some inherent difficulties with the language itself as well as due to lack of tools available to assist researchers. Therefore, the current project attempts to design and develop the modules of an Arabic QA system. For this purpose, the developed question answering system is based on a linguistic approach, using NooJ's linguistic engine in order to formalize the automatic recognition rules and then apply them to a dynamic corpus composed of medical journalistic articles. In addition, we present a method for analyzing medical questions (for a factoid questions). The analysis of the question asked by the user by means of the syntactic and morphological analysis.

The linguistic patterns (grammars) which allow us to extract the analysis of the question and the semantic features of the question of extracting the focus and topic of

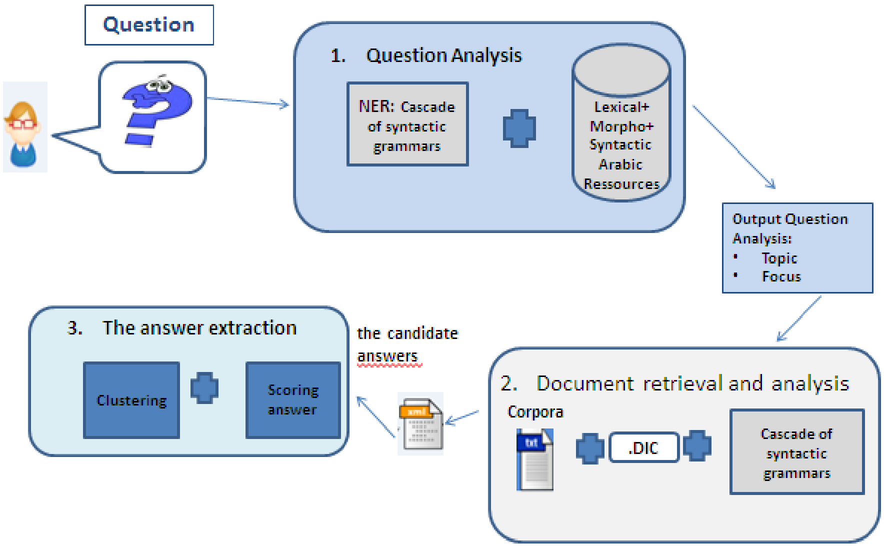

Question
1. Question Analysis
NER: Cascade of syntactic grammars
Lexical+ Morpho+ Syntactic Arabic Ressources
Output Question Analysis:
• Topic
• Focus
2. Document retrieval and analysis
Corpora
txt
.DIC
Cascade of syntactic grammars
the candidate answers
xml
3. The answer extraction
Clustering
Scoring answer

Fig. 1. Architecture of Question Answering System.

the question. In the next section, an overview of the state-of the art describes related works to question answering system. The section 3, we describe the generic architecture of the proposed QA system. In section 4, introduces our approach to annotation of medical factoid question and extraction of right answer.

2 State of Art

As explained in the introduction, Question-Answering systems present a good solution for textual information retrieval and knowledge sharing and discovery. This is reason why a large number of Q-A systems has been developed recently. The table below shows some work of question answering system by criterion.

After this investigation, to solve the problem of question answering system, the developed question answering system is based on a linguistic approach, using NooJ's linguistic engine in order to formalize the automatic recognition rules and then apply them to a dynamic corpus composed of Arabic medical journalistic articles.

The named entity recognizer (NER) is embedded in our question answering system in order to identify these answers and questions associated with the extracted named entities. For this purpose, we have adapted a rules-based approach to recognize Arabic named entities and right answers, using different grammars and gazetteer.

3 Architecture of Question Answering System

From a general viewpoint, the design of a QA system (Fig 1) must take into account three phases:

1. **Question analysis:** This module performs a morphological analysis to identify the question class. A question class helps the system to classify the question type to provide a suitable answer. This module may also identify additional semantic features of the question like the topic and the focus.
2. **Document retrieval and analysis:** The second motivation behind the question classification task is to develop the linguistic patterns for the candidate answers. These patterns are helpful in matching in parsing and identifying the candidate answers.
3. **The answer extraction:** This module selects the most accurate answers among the phrases in each corpus. The selection is based on the question analysis. The suggested answers are then given to the user as a response to his initial natural language query.

4 Our Approach

4.1 Named Entity Recognition (NER)

We think that an integration of a Named Entity Recognition (NER) module will definitely boost system performance. It is also very important to point out that an NER is required as a tool for al-most all the QA system components. Those NER systems allow extracting proper nouns as well as temporal and numeric expressions from raw text (Mesfar, 2007). In our case, we used our own NER system especially formulated for the Arabic medical domain. We have considered six proper names categories:

1. Organization: named corporate, governmental, or other organizational entity.
2. Location: name of politically or geographically defined location.
3. Person: named person or family.
4. Viruses: Names of medical viruses.
5. Disease: Names of diseases, illness, sickness.
6. Treatment: Names of Treatments.

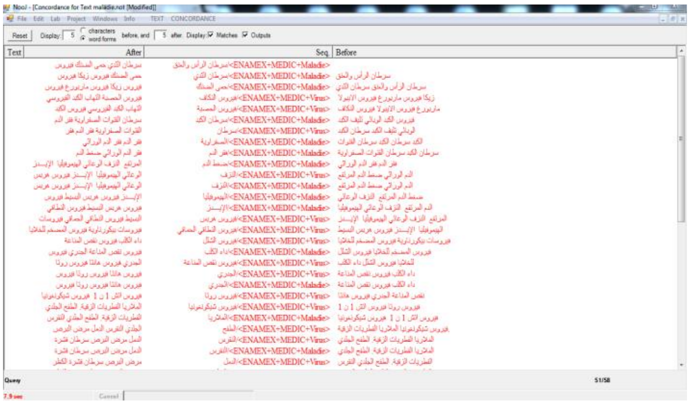
4.2 Automatic Annotation of Factoid Question in Standard Arabic

Our approach focuses on the problem of finding document snippets that answer a particular category of facts-seeking questions namely factoid questions. Simple interrogative questions which await an answer related to a named entity. The choice of factoid questions versus other types of questions is motivated by the following factors:

- Majority of the questions actually submitted to a search engine are factoid questions. Current search engines are only able to return links to full-length

Table 3. NER grammar experiments on our corpus.

Precision	**Recall**	**F-Measure**
0,90	0,82	0,88



NooJ - [Concordance for Text maladie.not [Modified]]

Display: 5 ○ characters ● word forms before, and 5 after Display: ☑ Matches ☑ Outputs

Text	After	Seq.	Before
	سرطان الثدي حمى الضنك فيروس	سرطان الرأس والعنق/<ENAMEX+MEDIC+Maladie>	
	حمى الضنك فيروس زيكا فيروس	سرطان الثدي/<ENAMEX+MEDIC+Maladie>	سرطان الرأس والعنق
	فيروس زيكا فيروس ماربورغ فيروس	حمى الضنك/<ENAMEX+MEDIC+Maladie>	سرطان الرأس والعنق سرطان الثدي
	فيروس الحصبة التهاب الكبد الفيروسي	فيروس الكلاف/<ENAMEX+MEDIC+Virus>	زيكا فيروس ماربورغ فيروس الايبولا
	التهاب الكبد الفيروسي فيروس الكبد	فيروس الحصبة/<ENAMEX+MEDIC+Virus>	ماربورغ فيروس الايبولا فيروس الكلاف
	سرطان القنوات الصفراوية فقر الدم	سرطان الكبد/<ENAMEX+MEDIC+Maladie>	فيروس الكبد الوبائي تليف الكبد
	القنوات الصفراوية فقر الدم فقر	سرطان/<ENAMEX+MEDIC+Virus>	الوبائي تليف الكبد سرطان الكبد
	فقر الدم فقر الدم الوراثي	الصفراوية/<ENAMEX+MEDIC+Maladie>	الكبد سرطان الكبد سرطان القنوات
	فقر الدم الوراثي ضغط الدم	فقر الدم/<ENAMEX+MEDIC+Maladie>	سرطان الكبد سرطان القنوات الصفراوية
	المرتفع النزف الوعائي الهيموفيليا الإيـــدز	ضغط الدم/<ENAMEX+MEDIC+Maladie>	فقر الدم فقر الدم الوراثي
	الوعائي الهيموفيليا الإيـــدز فيروس هربس	النزف/<ENAMEX+MEDIC+Virus>	الدم الوراثي ضغط الدم المرتفع
	الوعائي الهيموفيليا الإيـــدز فيروس هربس	النزف/<ENAMEX+MEDIC+Maladie>	الدم الوراثي ضغط الدم المرتفع
	الإيـــدز فيروس هربس البسيط فيروس	الهيموفيليا/<ENAMEX+MEDIC+Maladie>	ضغط الدم المرتفع النزف الوعائي
	فيروس هربس البسيط فيروس النطاقي	الإيـــدز/<ENAMEX+MEDIC+Maladie>	الدم المرتفع النزف الوعائي الهيموفيليا
	البسيط فيروس النطاقي الحماقي فيروسات	فيروس هربس/<ENAMEX+MEDIC+Virus>	المرتفع النزف الوعائي الهيموفيليا الإيـــدز
	فيروسات بيكورناوية فيروس المضخم للخلايا	فيروس النطاقي الحماقي/<ENAMEX+MEDIC+Virus>	الهيموفيليا الإيـــدز فيروس هربس البسيط
	داء الكلب فيروس نقص المناعة	فيروس الشلل/<ENAMEX+MEDIC+Virus>	فيروسات بيكورناوية فيروس المضخم للخلايا
	فيروس نقص المناعة الجدري فيروس	داء الكلب/<ENAMEX+MEDIC+Maladie>	فيروس المضخم للخلايا فيروس الشلل
	الجدري فيروس هانتا فيروس روتا	فيروس نقص المناعة/<ENAMEX+MEDIC+Virus>	للخلايا فيروس الشلل داء الكلب
	فيروس هانتا فيروس روتا فيروس	الجدري/<ENAMEX+MEDIC+Virus>	داء الكلب فيروس نقص المناعة
	فيروس هانتا فيروس روتا فيروس	الجدري/<ENAMEX+MEDIC+Maladie>	داء الكلب فيروس نقص المناعة
	فيروس اتش 1 ن 1 فيروس شيكونغونيا	فيروس روتا/<ENAMEX+MEDIC+Virus>	نقص المناعة الجدري فيروس هانتا
	الملاريا الفطريات الزقية الطفح الجلدي	فيروس شيكونغونيا/<ENAMEX+MEDIC+Virus>	فيروس روتا فيروس اتش 1 ن 1
	الفطريات الزقية الطفح الجلدي النقرس	الملاريا/<ENAMEX+MEDIC+Maladie>	فيروس اتش 1 ن 1 فيروس شيكونغونيا
	الجلدي النقرس الدمل مرض البرص	الطفح/<ENAMEX+MEDIC+Virus>	فيروس شيكونغونيا الملاريا الفطريات الزقية
	الدمل مرض البرص سرطان قشرة	النقرس/<ENAMEX+MEDIC+Virus>	الملاريا الفطريات الزقية الطفح الجلدي
	الدمل مرض البرص سرطان قشرة	النقرس/<ENAMEX+MEDIC+Maladie>	الملاريا الفطريات الزقية الطفح الجلدي
	مرض البرص سرطان قشرة الكظر	الدمل/<ENAMEX+MEDIC+Virus>	الفطريات الزقية الطفح الجلدي النقرس

Query 51/58

7.9 sec

Fig. 4. Result of NER NooJ syntactic grammar.

documents rather than brief document fragments that answer the user’s question.

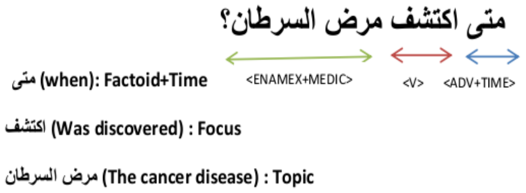
- The frequent occurrence of factoid questions in daily usage is confirmed by the composition of the question test sets in the QA track at Text Retrieval Conference.
- Most approaches of QA use NER as a foundation for detecting candidate answers.

As far as current research is concerned, our QA module accepts, as input, only Arabic factoid questions. Then, in order to look for the best answer, it gives the maximum amount of information (syntactic, semantic, distributional, etc.) from the given question, such as the expected answer the focus and topic of the question. This information will play an important role in the phase of extraction candidate answers.

- **Topic:** the topic corresponds to the subject matter of the question.
- **Focus:** the focus corresponds to the specific property of the topic that the user is looking for.

The following example shows the detailed annotation of the identified parts of a question. Example:

- When was cancer discovered?



متى اكتشف مرض السرطان؟

متى (when): Factoid+Time

اكتشف (Was discovered) : Focus

مرض السرطان (The cancer disease) : Topic

5 Experiments and Results

5.1 Named Entity Recognition (NER)

5.1.1 Evaluation

To evaluate our NER local grammars, we analyse our corpus to extract manually all named entities. Then, we compare the results of our system with those obtained by manual extraction. The application of our local grammar gives the following result:

According to these results, we have obtained an acceptable identification of named entities. Our evaluation shows F-measure of 0.88. We note that the rate of silence in the corpus is low, which is represented by the recall value 0,88 because journalistic texts of our corpus are heterogeneous and extract-ed from different resources.

5.1.2 Discussion

Despite the problems described above, the used techniques seem to be adequate and display very encouraging recognition rates. Indeed, a minority of the rules may be sufficient to cover a large part of the patterns and ensure coverage. However, many other rules must be added to improve the recall.

5.2 Automatic Annotation of Factoid Question in Standard Arabic

5.2.1 Evaluation

To evaluate our automatic annotation question local grammars, we also analyze our user's queries to extract manually the question analysis. Then, we compare the results of our system with those obtained by manual extraction. The application of our local grammar gives the following result:

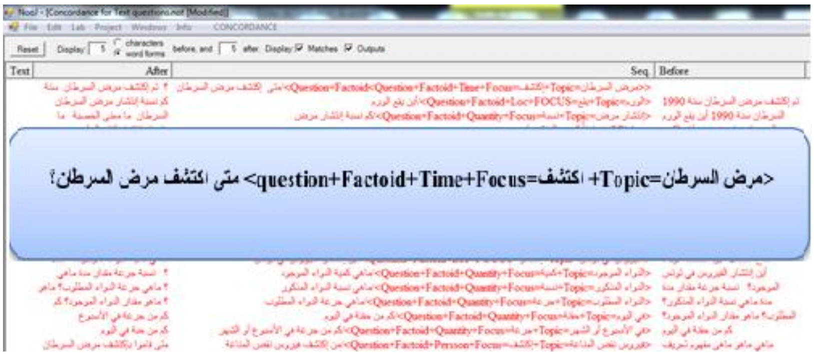
According to these results, we have obtained an acceptable annotation of question. Our evaluation shows F-measure of 0.73. We note that the rate of silence in the corpus is low, which is represented by the recall value 0.72. This is due to the fact that this assessment is mainly based on the results of the NER module.

5.2.2 Discussion

Errors are often due to the complexity of user’s queries sentences or the absence of their structure in our system In fact, the Arabic sentences are usually very long, which sets up obstacles for the question analysis. Despite the problems described above, the developed method seems to be adequate and shows very encouraging extraction rates. However, other rules must be added to improve the result.

Table 4. Annotation question grammar experiments.

Precision	**Recall**	**F-Measure**
0,75	0,72	0,73


NooJ - [Concordance for Text questions.not [Modified]]
File Edit Lab Project Windows Info CONCORDANCE
Reset
Display 5 ○ characters ● word forms before, and 5 after
Display ☑ Matches ☑ Outputs
Text
After
Seq.
Before

<مرض السرطان=Topic+ اكتشف=question+Factoid+Time+Focus> متى اكتشف مرض السرطان؟

Fig. 5. Result of Annotation NooJ syntactic grammar.

6 Conclusion

Arabic Question Answering Systems could not match the pace due to some inherent difficulties with the language itself as well as upon to the lack of tools offered to support the researchers. The task of Question Answering can be divided into three phases:

- Question analysis,
- Document retrieval,
- Analysis, and answer extraction.

Each of these phases plays crucial roles in overall performance of the Question Answering Systems. As a future work, we work on the two other phases of Question Answering Systems; that is “Answer Extraction” and “Document Analysis”. In Document Analysis, we can look for such methods used in information retrieval including tools, evaluation, and corpus. In Answer Analysis, we can look for such methods used in this phase including evaluation, tools, and corpus. Finally, as a long-term ambition, we intend to consider studying the processing of the “why” and “how” question types.

References

1. Benajiba, Y., Rosso, P., Lyhyaoui, A.: Implementation of the ArabiQA Question Answering System's components. In: Proc. Workshop on Arabic Natural Language Processing. In: 2nd Information Communication Technologies Int. Symposium, ICTIS-2007, Fez, Morroco (2007)

2. Ben A. F.: Un système de Question Réponse coopératif sur le web. Thèse de doctorat de l'université Paul Sabatier (2014)
3. Chu-Carroll, J., Krzysztof, C., Prager, J., G., S. B.: IBM's PIQUANT II in TREC2004. In: TREC-13 (2002)
4. Doaa S., Moreno-Sandoval, A., Bueno-Díaz, C., Garrote-Salazar, M., Guirao, J. M.: Medical term extraction in an Arabic medical corpus. In: Proceedings of LREC'12 (2012)
5. Ferret, O., Grau, B., Illouz, G., Jacquemin, C., Masson, N.: QALC - the Question Answering Program of the Language and Cognition Group at LIMSI-CNRS. In: Proceedings of the 8th Text Retrieval Conference, NIST Special Publications, pp. 465–475 (1999)
6. Hammou, B., Abu-Salem, H., Lytinen, S., Evens, M.: A Question answering system to support the Arabic language. In: Proc. of the workshop on Computational approaches to Semitic languages, ACL, pp. 55–65 (2002)
7. Laurent, D., Séguéla, P., Nègre, S.: Cross lingual question answering using Qristal for Clef. In: Working Notes (2005)
8. Mohammed, F. A., Nasser, K., Harb, H. M.: A knowledge-based Arabic Question Answering System (AQAS). In: ACM SIGART Bulletin, pp. 21–33 (1993)
9. Moldovan, D., Harabagiu, S., Pasca, M., Mihalcea, R., Girju, R., Goodrum, R., Rus, V.: The Structure and Performance of an Open-Domain Question Answering System. In: Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, pp. 563–570 (2002)
10. Nyberg, E, Mitamura, T.: Evaluating QA systems on multiple dimensions (2002)
11. Plamondon, L., Kosseim, L., Lapalme, G.: The quantum question answering system at trec-11. In: Proceedings of the Eleventh Text Retrieval Conference (TREC-2002), pp. 750–757 (2002)
12. Neifar, W., Ben-Ltaief, A.: Acquisition terminologique en arabe: État de l'art, Actes de la conférence conjointe JEP-TALN-RECITAL (2016)
13. Silberztein, M.: NooJ manual. Available at the WEB site http://www.nooj4nlp.net (2003)
14. Silberztein M.: NooJ's Linguistic Annotation Engine. INTEX/NooJ pour le Traitement Automatique des Langues, Cahiers de la MSH Ledoux. Presses Universitaires de Franche-Comté, pp. 9–26 (2006)

ISSN 1870-4069

A Rule-Based Approach for Arabic Named Entity Metonymy Resolution

Sadek Mansouri[1], Chahira Lhioui[1], Mounir Zrigui[1], Mbarek Charhad[2]

[1] LATICE Laboratory,
Research Department of Computer Science,
Tunisia

[2] Taibah University,
Saudi Arabia

mansouri sadek@hotmail.fr, chahira_m1983@yahoo.fr,
mounir.zrigui@fsm.rnu.tn, mbarek.charhad@gmail.com

Abstract. Named entity metonymy resolution aims to determine the correct meaning and type of named entity in a given context. This task has recently been subject to a growing interest for the Natural Language Processing community and mainly for Arabic language. In this paper, we have developed a new method to solve arabic named entity metonymy in text news. Our main contribution consists to propose a robust parser that extracts metonymy patterns of each entity using a set of contextual rules based on part-of-speech tags and semantic triggers. The performance of our proposed approach has been evaluated on a relatively large size corpus. The obtained results are promising and improve the named entity recognition system.

Keywords: Metonymy resoution, contextual information, arabic NE recognition.

1 Introduction

Everyday, the mass of stored information is becoming colossal and continuing to grow exponentially. Information is conveyed through different modes of communication such as video, audio and text. In this heterogeneous environment, the challenge is to facilitate user's fast and relevant access to the desired information, without being confused by the huge amount of data that is offered to him. Many Automatic Natural Language Processing (NLP) applications interested in developing methods and tools to respond to this challenge, such as the extraction of information, the search for information, indexing and automatic translation.

In these different domains, the task of named Entities Recognition(NER) plays a transversal role. The concept of named entity appeared in the mid-1990s as being a subtask of the information extraction activity. It consists of identifying certain textual objects such as person name, organization name and location. Over the years, research on these linguistic units focused on increasingly complex issues like disambiguation and enriched annotation but also on their recognition in different contexts.

pp. 53–62; rec. 2018-01-16; acc. 2018-03-05

Despite the performance of proposed approaches, the recognition of named entities is still a difficult task especially for the Arabic language. This is due to the specific features of Arabic text [1–3]:

- Lack of capitalization: Unlike Indo-European languages, Arabic language does not have the concept of capitalization. This represents a major obstacle for the Arabic language during the extraction of NE. In fact, capital latter is very effective in the proper names recognition process for certain languages as English or French. So its absence in the Arabic language imposes an urgent necessity to find alternatives and, ultimately, to use other conventional means such as lexicons, triggers words and grammatical rules.
- Complicated morphology: Arabic is a highly-inflected language. It uses an agglutinative strategy to form a word. If NE appears with agglutinative form, then this poses a difficulty for the identification of this entity.
- The absence of vowels: A non-vowelized word has many ambiguities in meaning or syntactic function. For example, the word can be a verb (go) to a voyellation or a proper name (Gold) for another. Thus, this example illustrates the impact of the vowels lack in words recognition.

In this work , we are particularly interested to solve the problem of metonymy of Arabic named entities. As a definition metonymy is a linguistic operation that allows the use of one entity to represent another which, as a result, leads to the emergence of polysemy phenomena for lexical units. Metonymy corresponds more exactly to a case where one entity seems to refers to many different NEs types which lead to an semantic ambiguity as shown in the following examples.

تصفيات مونديال 2018: **فرنسا** تتصدر المجموعة الأولى

Example 1. “FIFA World Cup Qualifiers 2018: **France** lead the first group”.

النادي الثقافي **محمود المسعدي** بتازركة

Example 2. “Cultural Club **Mahmoud Masadi** Btazarka”.

In the first example *France* refers to sport club not a place or country while in the second sentence *Mahmoud Masadi* refers to name of cultural club not to name of person. This type of ambiguity makes identification of NEs more difficult and raises NE disambiguation problem as one of the main challenges to research not only in the semantic web but also in areas of natural language processing in general. In this paper, we adress these challenges to solve the problem of metonymy for arabic named entity in text news.

To reach our objective, we define a several regular metonymic patterns for each entity class. We also have to establish a set of syntactic rules which provide to detect these patterns in given text and implemented with the linguistic platform Farassa. The rest of this paper is organized as follows: Section 2 provides insights from literature on related work. Section 3 presents the proposed approach. Section 4 discusses the approach evaluation and results, and Section 5 concludes the paper.

2 Related Works

The first work developed in the metonymy resolution (MR) task comes from the evaluation company SemEval 2007 [1] [4] and after by [5]. In this last work, the set of descriptors has been updated to include: the grammatical role of the potential metonymic word (PMW) (as subj, obj), determinant of PMW, grammatical number of PMW (singular, plural), number of words in PMW and the number of grammatical roles of PMW in a given context. The system proposed by [6] achieved the best results in terms of accuracy 85.2% using these features and a maximum entropic classifier.

In the same context, [7] reaches an accuracy of 85.1% using local characteristics of syntactic and global distribution generated with a suitable Xerox and proprietary deep analyzer. This system is based on an unsupervised approach with the use of contextual syntactic similarity calculated on large corpora such as the British National Corpus (BNC) with 100M tokens.

[8] use SVM (Support Vector Machine) with new descriptors (in addition to the descriptors provided by[5]) integrating grammatical collocations extracted from the BNC to learn selection preferences. Moreover, their system is based on external linguistic resources like WordNet 3.0, the Wikipedia category for optimization of the global context. In SemEval 2007, this system achieves an accuracy of 86.1%. Recently, [9] use a minimalist neural approach combined with a novel predicate window method for metonymy resolution.

Additionally, their system contribute with a new Wikipedia-based MR Dataset called RelocaR, which is tailored towards locations as well as improving previous deficiencies in annotation guidelines. The proposed system achieve 84.8% for the SemEval dataset. We remark that all proposed systems are designed to French and English language and to the best of our knowledge, our work present the first attempt to solve the problem of metonymy of arabic named entity.

3 Proposed Method

Figure 1 shows the global architecture of our proposed system. This system include three stages: Morpho-syntactic analysis, NE recognition step which integrates a divers set of lexical resources such as gazetteers and syntactic grammar. The final stage focuses on metonymy resolution to assign the correct type of an ambiguous named entity and enhance the initial results of NE recognition.

3.1 Morpho-Syntactic Analysis

In this phase, we segment input text into words based on spaces delimiter. Then we proceed by a morphological analysis of the corpus to extract useful information that will be used in the named entity recognition system. Given the agglutinate structure that has the majority of Arabic words, our morphological analyzer can separate and identify morpheme and associate all information necessary to the current treatment.

[1] http://nlp.cs.swarthmore.edu/semeval/tasks/index.php

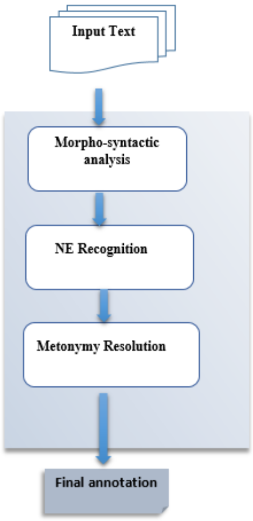


Fig. 1. The proposed system.

These forms are decomposed to recognize affixes (conjunctions, prepositions, pronouns, etc.) attached to it. These morphological possibilities in this analyzer can facilitate the identification of trigger words. As a matter of fact, each of these forms is associated with a set of useful linguistic information for the following step: lemma, grammatical labels, gender and number, syntactic information (such as: + Transitive) distributional information (such as: + Human), etc.

3.2 Named Entities Recognition

In the second step, we parse text files to detect named entities using Farassa tool[2]. Farasa is a fast and accurate text processing toolkit for Arabic named entities based on the CRF++[3] implementation of CRF sequence labeling. To improve NER in our datatset. We built a large gazetteer from Wikipedia. The gazetteer had 80,908 locations, 36,391 organizations, and 91,880 persons and 2000 event.

3.3 Metonymy Resolution

The task of metonymy resolution implies identifying the correct interpretation of a named entity in a given context. Some entities can be hard to annotate because of ambiguity between main types, such as locations, GPEs and organizations. Such entities can often take on different roles, according to their usage. The ACE guidelines describe two forms of metonymy.

[2] http://qatsdemo.cloudapp.net/farasa/demo.html

[3] http://code.google.com/p/crfpp/

Nickname metonymy occurs when the name of one entity is used to refer to another entity, such as a capital city referring to a government, or a location name denoting asports team. Cross-type metonymy occurs when multiple aspects of an entity are referenced at the same time, such as organizations and the facilities they occupy (e.g. They will be visiting the White House tomorrow). In this work, we intressed to metnoymy resolution. To this end, we first describe a serval regular metonymic patterns for each entity class and then we propose a syntactic grammars aiming to detect these patterns in arabic text news and solve the metnoymy problem.

Metonymic Patterns. In this section , we present a set of metonymic patterns which used to solve named entities ambiguity .

- Metonymic patterns of Organisation class
  1. **Organisation-to-event**:
  This Annotation is to be used when the name of the organization refers to an event organized by this organization.

     المؤتمر الدولي **لمنظمة الأمم المتحدة**

     "International Conference of **the United Nations**".
  2. **Organisation-to-Person**:
  This annotation is to be used when an organization name refers to people associated with it.

     الناطق الرسمي **لمنظمة الدفاع عن حقوق المستهلك**

     "The official spokesman of **the Organization for the Defense of Consumer Rights**".
  3. **Organisation-to-product**:
  Another widely used metonymy is Organisation-to-product, where the name of a commercial organization refers to its products.

     سيارات **فورد** الشعبية تقتحم السوق العالمية

     "**Ford's** popular cars are breaking into the global market".
  4. **Organisation-to-location**:
  This annotation is to be used when a company name referred to its own places.

     أنفاق المقاومة الفلسطينية في قطاع غزة

     "**Palestinian resistance tunnels** in the Gaza Strip".
- Metonymic patterns of location class similarly to organization names, place names can also refer not only to their primary reference, but also to other references related to it. The most frequent type is :
  1. **location-to-event**:
     We use this annotation when the country name refers to an event having taken place in this country.

الدورة العاشرة لأيام **قرطاج** السينمائية

“10th **Carthage** Film Festival”.

2. **location-to-Organization**:
   This annotation is to be used when a country name refers to people or organizations associated with it. We can think of when it comes to the government, a sports team or of the country’s population.

   تونس تترشح لمونديال 2018

   “**Tunisia** to run for 2018 World Cup”.

   مجلس النواب يصادق على قانون المالية

   “**The House of Representatives** approves the Finance Act”.

- Metonymic patterns of Person class
  1. **Person-to-time**
     This annotation is to be used when a person’s name refers to a period of time.

     مصطفي الفيلالي وزير زمن **بورقيبة**

     “Mustafa Filali Minister of Time of **Bourguiba**”.
  2. **Person-to-location**
     This annotation is to be used when a person name referred to location.

     المدرسة الإعدادية **على بلهوان**

     “Preparatory School **Ali Belhwan**”.
  3. **Person-to-Organization**
     We apply this pattern when a person name referred to Organization.

     جمعية **رحمة** لرعاية الأيتام بمنوبة

     “**Rahma** Association for Orphans Care in Manouba”.

Elaboration of Contextual Rules: Our focus in this part of study is to identify the contextual rules for each pattern. To this end, we have started to carry out a study of the training corpus. For all types of named entities, our focus was on the one hand, the types of grammatical relationships in which the units were involved. on the other hand, the lexical or contextual information attached to the arguments of these relations.

For each metonymy pattern, we therefore analyzed all of its occurrences (attached to a lexical unit) in the training corpus to derive grammatical and lexical configurations playing the role of metonymy resolution. This study is completed by the elaboration of contextual rules reflecting the discriminant configurations for each pattern. For example, let’s take the following rule:”if the country name is followed by an verb so the pattron location-to-organization should be applied”. The second contextual rule of location pattern is: “if the country name proceed by trigger reltad to event such as (Symposium, Forum, Conference), pattern of organization-to-event should be applied”.

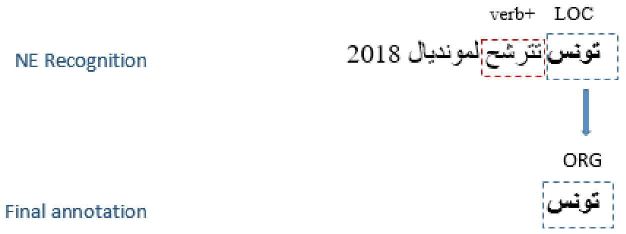


Fig. 2. Example for location-to-organization rule.

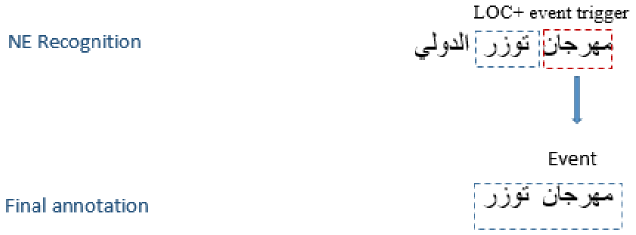


Fig. 3. Example for location-to-event rule.

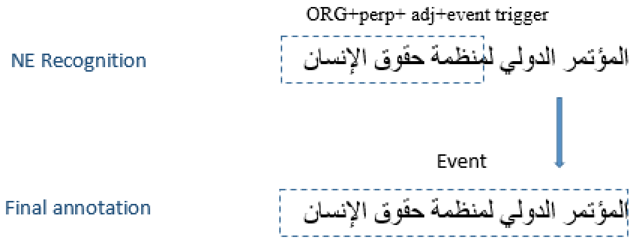

ORG+perp+ adj+event trigger
NE Recognition
المؤتمر الدولي لمنظمة حقوق الإنسان
Event
Final annotation
المؤتمر الدولي لمنظمة حقوق الإنسان

Fig. 4. Example for Organization-to-event rule.

For the Organization class, the most discriminating indices are prepositional phrases such as International Conference of the United Nations .To solve the problem , we apply this contextual rule :”if the country name proceed by trigger related to event such as (Symposium, Forum, Conference) + adj + preposition, pattern of organization-to-event should be applied”.

For person class, the main problem of metonymy is caused by the fusion of meaning with other named entities. To solve this, we define this contextual rule: “if the name of person proceed by trigger related to location, so, we apply Person to location pattern.”

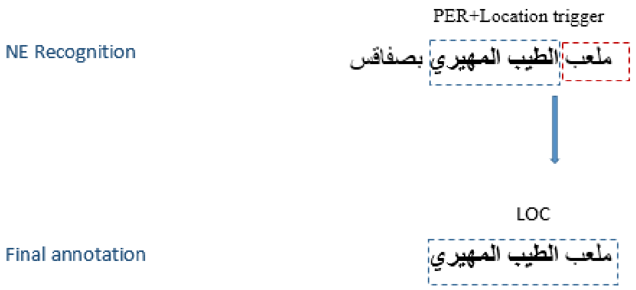


Fig. 5. Example for Person-to-location rule.

Table 1. Distribution of metonymy patterns.

Entity	**Metonymy Petterns**	**%**
Organization	Organization-to-event	7%
	Organization to Person	3%
	Organization-to-Location	5%
Location	location-to-event	3%
	location-to-organisation	12.3 %
Person	Person-to-location	5.6%
	Person-to-organization	12.3%
Total		37.6%

4 Evaluation and Results

4.1 Corpus

To the best of our knowledge, there is no reference benchmark dataset for Arabic named entities disambiguation However, to evaluate our approach we have assembled a large dataset on news articles extracted form Arabic Wikinews covering different domains of technology, politics and sport. Table 1 show the distribution of metonymic patterns in the training corpus and gives in idea about the complexity of the metonomy resolution task .It presents the number of metonymy cases for each named entity. In this table, we find that the total cases is equal to 37.6%.

4.2 Results

The evaluation of our system is as follows:

- The texts of evaluation corpus are all manually annotated.
- After we have automatically annotated these texts using our named entities recognition system.
- Then we have applied our method of metonymy resolution to enhance the initial results.

Table 2. Annotation results.

Entity	without (MR)	with (MR)
Organization	78,90%	85,5%
Person	81,25%	82,5%
Location	81,91%	83%

In the final step ,we have established a comparison between these annotations with corpus quality assurance. The overall evaluation of entities extracted by our system is based on the use of F-measure. This measure combines the precision and the recall. Precision measurement is defined by the percentage of entities found by the system and which are correct. The recall is defined as the ratio between the numbers of found correct entities by the number of entities extracted from the reference articles.

Table 2 shows the obtained results for different entities with and without metonymy resolution (MR).We remark that our proposed method(MR) improve the f measure rate of the system by 10.05 % particularly organization entity achieved the best result.This due to the strong presence of metonymy patterns of Person to Organization and Person to Organization in dataset and the performance of our proposed syntactic grammar.

5 Conclusion

The treatment of lexical metonymy particularly the metonymy of named entities, can improve a number of treatments, among which the extraction tasks information and questions and answers. Our work present the first attempt to solve the problem of metonymy for Arabic named entities. The main contribution is to define the metonymy patterns and syntactic grammar aiming to find the exact type of each entity according to the context where is used. The experimentation and evaluation results are promising. In future works, we can test our system on a larger corpus and try to define new metonymy patterns to improve our proposed approach.

References

1. Mansouri, S., Lhioui C., Charhad, M., Zrigui, M.: Text-to-Concept: A Semantic Indexing Framework for Arabic News Videos. In: Gelbukh A. (eds) Computational Linguistics and Intelligent Text Processing. CICLing 2017 (2018)
2. Lhioui, C., Zouaghi, A., Zrigui, M.: A Rule-based Semantic Frame Annotation of Arabic Speech Turns for Automatic Dialogue Analysis, Procedia Computer Science, vol. 117, pp. 46–54 (2017)
3. Lhioui, C., Zouaghi, A., Zrigui, M.: Realization of Minimum Discursive Units Segmentation of Arab Oral Utterances. International Journal of. Computational Linguistics and Applications. vol. 7, no. 1, pp. 31–50 (2016)
4. Markert, K., Nissim, M.: SemEval-2007 Task 08: Metonymy Resolution at SemEval-2007, pp. 36–41 (2007)
5. Markert, K., Nissim, M.: Data and models for metonymy resolution. Lang. Resour. Evaluation, vol. 43, no. 2, pp 123–138 (2009)

6. Farkas, R., Simon E., Szarvas, G., Varga, D.: GYDER: Maxent Metonymy Resolution. pp. 161–164 (2007)
7. Brun. C., Ehrmann, M., Jacquet, G.: A Hybrid System for Named Entity Metonymy Resolution. LTC 2007: pp. 118–130 (2007)
8. Nastase, V., Strube, M.: Combining Collocations, Lexical and Encyclopedic Knowledge for Metonymy Resolution. EMNLP 2009: pp. 910–918 (2007)
9. Gritta, M., Taher-Pilehvar, M., Limsopatham, N., Collier, N.: Vancouver Welcomes You! Minimalist Location Metonymy Resolution. vol. ACL, no. 1, pp. 1248–1259 (2017)
10. Mahmoud, A., Zrigui, M.: Semantic Similarity Analysis for Paraphrase Identification in Arabic Texts. PACLIC 2017, pp. 274–281 (2017)
11. Zouaghi, A., Zrigui, M., Antoniadis, G.:Compréhension automatique de la parole arabe spontanée. Traitement Automatique des Langues, (2008)
12. Ayadi, R., Maraoui, M., Zrigui, M.: Latent topic model for indexing arabic documents. International Journal of Information Retrieval Research (IJIRR), vol. 4, no. 2, pp. 57–72 (2014)
13. Merhben, L., Zouaghi, A., Zrigui, M.: Lexical Disambiguation of Arabic Language: An Experimental Study Polibits, pp. 49–54 (2012)
14. Mohamed, M., Mallat, S., Nahdi, M. A., Zrigui, M.: Exploring the potential of schemes in building NLP tools for Arabic language. International Arab Journal of Information Technology (IAJIT), vol. 12, no. 6 (2015)
15. Mansouri, S., Charhad, M., Zrigui, M.: Arabic Text Detection in News Video Based on Line Segment Detector. Research in Computing Science. vol. 132, pp. 97–106 (2017).
16. Slimi, J., Mansouri, S., Ben-Ammar, A., Alimi, A. M.: Semantic browsing in large scale videos collection. OAIR 2013, pp. 53–56 (2013).
17. Slimi, J., Mansouri, S., Ben-Ammar, A., Alimi, A. M.: Video exploration tool based on semantic network. OAIR 2013, pp. 213–214

ISSN 1870-4069

Customer Feedback Analysis Using Machine Learning Techniques

Anupam Jamatia, Kanishka Joshi, Kundan Kumar,
Shivam Kumar, Subrito Haldar

Department of Computer Science & Engineering,
National Institute of Technology Agartala,
India

{anupamjamatia, kanishkj34, kundanjnv11,
shivam.nita14, subrito996}@gmail.com

Abstract. Customer feedback are the representation of customers' opinions about the concerned products in today's business organization. Thus its analysis is essential in providing a company insights into what it has to do to render better customer experience in the future. Our work focuses on the automatic processing of customer feedback using machine learning approaches and subsequently analyzing them, which is otherwise an impossible task to do manually due to customer experience on sheer volume and variety of services, products. We compare the performance of different machine learning classifiers such as Naïve Bayes, Support Vector Machine, k-Nearest Neighbors and Random Forest on the collected corpus. The maximum accuracy is obtained using Random Forest classifier with an accuracy of 69.74%.

Keywords: Customer feedback, machine learning.

1 Introduction

The purpose of the customer feedback is that it provides marketers and business owners with insight that they can use to improve their business, products and overall customer experience. Classification of feedback is essential to gain the better perspective on the views of the customers. Customer feedback analysis measures how happy customers are with a company's products and services. With the ever-increasing size of feedback data, it has become an improbable task to manually inspect each review.

So it is necessary to automate the overall process to provide businesses with a better view of what it has to change, what it has to improve on, and what it has to do, to retain and grow revenue and profit. Customer feedback analysis has become an industry on its own. Hence many companies understandably want to automate customer feedback analysis system but a major hurdle is to deal with multilingual environment which exists in all over the world. There are several online survey-based companies who acquire customer data from their clients and do the analysis. Firstly, some commonly used categorizations include five-class viz. 'Excellent'- 'Good'- 'Average'- 'Fair'- 'Poor' by Yin *et. al* [20] and SurveyMonkey[1].

[1] https://www.surveymonkey.com/

pp. 63–72; rec. 2018-01-16; acc. 2018-03-05

Secondly, there are opinion and responsiveness based five-class viz. 'Positive'- 'Neutral'- 'Negative'- 'Answered'- 'Unanswered' by Freshdesk[2]. Lastly, there are seven-class 'Refund'- 'Complaint'- 'Pricing'- 'Tech Support'- 'Store Locator' - 'Feedback' - 'Warranty' by Sift[3]. These surveys are a vital tool for a variety of research fields, including marketing, social and official statistics research. A lot of work has been done in the field of sentiment analysis of feedback that classifies the sentiment polarity of the customer feedback into positive, negative, neutral and so on.

But interpretation of those reviews into bug, complaint, comment or request for better customer support is essential as well. Our work deals with classifying a customer review (in English) into one or more of the six predefined classes taken from Liu *et. al.* [13]. The classes are 'comment', 'request', 'bug', 'complaint', 'meaningless' and 'undetermined'. This paper can be viewed as multiclass classification [12, 21, 17, 4, 14] problem. We have used TF-IDF feature vectors and then used supervised machine learning techniques to train our dataset and subsequently test it.

The rest of the paper is organized as follows; in Section 2 we discussed the related research work on the customer feedback analysis. Data collection and preprocessing are described in Section 3. In Section 4, the description of the features used are given and the performance of four different machine learning methods are presented. Results, observations and error analysis are discussed in Section 5 and Section 6 respectively. Section 7 sums up with future research scope. The source code of our system can be found here.[4]

2 Related Work

There has been some significant work done in the area of customer feedback analysis, sentiment analysis of feedback and multiclass classification of feedback. For example, the work by Bentley and Batra [1] on Microsoft Office feedback describes how an engineer or a manager finds the signal in feedback to make business decisions by using classification, on-demand clustering and other machine learning techniques. The problem of sentiment polarity categorization has been tackled in Fang and Zhan [5].

In their experiment, random forest model performs the best on manually-labeled and machine-labeled sentences in case of sentence-level categorization but Support Vector Machines (SVM) model and Naïve Bayesian model perform better than Random Forest model in case of review-level categorization. Large feature vectors in combination with feature reduction can train linear support vector machine can achieve high classification accuracy, which was described by Gamon [6] in his paper on sentiment classification on customer feedback.

The paper suggests that the addition of deep linguistic analysis features to a set of surface level word n-gram features contributes consistently to classification accuracy in this domain. Mukherjee and Bhattacharyya [16] in their paper regarding feature specific sentiment analysis for product reviews have used dependency parsing method to identify relations among the opinion expressions.

[2] https://freshdesk.com/

[3] https://www.startupranking.com/sift

[4] https://drive.google.com/drive/folders/1d0w0yRbubqHC4R7ev7KilU3gybsDscDQ

Other such related research includes the paper by Chakankar *et al.* [3] which constitutes sentiment analysis of users' reviews and comments. They have used three different datasets and have classified the reviews/ comments as being positive or negative.

- The first dataset has movie reviews from IMDB [5]. They have used 25000 highly polar reviews for training purpose and 25000 reviews for testing purpose. For this dataset SVM model has obtained the best accuracy of 88.89%.
- The second dataset has 2000 processed movie reviews drawn from IMDB archive. For this dataset also, SVM model has outperformed other classifiers.
- The third dataset consisted of social commentary having insults; out of 3947 instances of social commentary 1049 are insults. For this dataset, Naïve Bayesian model has outperformed SVM model and Logistic Regression model.

A set of techniques has been proposed by Hu and Liu [8] to mine and summarize reviews based on data mining and natural language processing methods which is useful to common shoppers as well as product manufactures. They have performed the task in three steps.

- Mining product features that has been commented by the customers.
- Identifying opinion sentences in each reviews and deciding whether each opinion sentence is positive or negative.
- Summarizing the results by aggregating the results from the previous steps.

Two-class sentiment classification of movie reviews as positive or negative using machine learning techniques has been done by Pang *et al.* [18]. They have used Naïve Bayes, maximum entropy classification and SVM. They have taken n-grams as feature and SVM gives the best accuracy of 82.9% on unigrams feature.

3 Dataset

We received the dataset from the IJCNLP 2017 organizers of shared tasks for customer feedback analysis[6]. Each document in the dataset was pre-annotated into one of the classes, with a few documents (4.5%) being classified into more than one class. In the corpus, there are a total number of 3723 documents, which are distributed into six predefined classes, namely comment, request, bug, complaint, meaningless and undetermined. A few samples have been listed in table 1.

From them, 'comment' and 'complaint' classes have the maximum number of feedback. The class 'undefined' has the least number of feedback. About 4.5% of the feedback were annotated with multiple labels. The entire distribution of dataset into classes has been displayed in table 2.

[5] http://www.imdb.com

[6] https://sites.google.com/view/customer-feedback-analysis/

Table 1. Samples of feedback sentences from the dataset.

Statement	Qualifier
It is so wonderful to use.	Comment
Being a new Apple Developer, I needed a fast program that would work fast and has an easy User Interface.	Request
Phone froze as if the app had a virus.	Bug
Beautiful afternoon at the Bristol!	Meaningless
Even the accessories in the app look fake.	Complaint
Maybe old style clothing too from civil war era not just city slicker clothing.	Undetermined
It's nothing but it consumes a large amount a CPU and memory.	Complaint, Bug

Table 2. Class distribution in corpus.

Class	Number of feedback	Number of tokens
Bug	92	1553
Comment	2034	22099
Complaint	1096	15720
Meaningless	354	3600
Request	122	1827
Undefined	25	336
Total	3723	45135

The data provided by IJCNLP Shared Task 2017 organizer was raw in nature. That is, extra data (meta data) was present along with it. The raw data was in the format of: Raw Data = Text ID + Sentence + Classifier. Hence, pre-processing of raw data was necessary. First, the 'Text ID' was removed. Afterwards, stop-words, that is, common words which would appear to be of little value in helping select documents matching a user need, are excluded.

Further, words with frequency of exactly one were also removed, as they do not contribute to the overall classification process as well. Later the data was tokenized. We have used NLTK[7] sentence tokenizer for tokenizing sentences and then used NLTK word tokenizer for tokenizing words. After the above process, we have got refined data.

4 Experiments

We have used some of the popular supervised machine learning algorithms in our approach. We have used TF-IDF as features of the corpus to convert the textual representation of information into vector space model. Thereafter we divided the vector space into training and testing data using k-fold algorithm (k=10). Subsequently we implemented six classifiers, namely Gaussian Naïve Bayes classifier, Multinomial Naïve Bayes classifier, Bernoulli Naïve Bayes classifier, SVM, k-Nearest Neighbours (k-NN) classifier and Random Forest classifier. We then calculated the precision and accuracy score for each and compared them.

[7] http://www.nltk.org/api/nltk.tokenize.html

We analyzed the results with the help of confusion matrix. After pre-processing of the data, we carried out feature selection and performed an analysis using TF-IDF.

$$\text{TF}(\text{word}) = F_{\text{count}}(\text{word})/N, \tag{1}$$

$$\text{IDF}(\text{word}) = log_e(N/E_{\text{count}}(\text{word})), \tag{2}$$

$$\text{TF} - \text{IDF} = \text{TF} \times \text{IDF}. \tag{3}$$

At first, we calculated the Bag-of-Words vector and using the same, we calculated the term frequency (TF) and later inverse document frequency (IDF) values for each unique word in each of the documents. Following that a 2-dimensional vector space was created.After the production of feature vectors, we then created the training and testing set using k-fold cross validation [11] algorithm, setting k=10, i.e. 90% of the dataset was kept in the training set and the remaining 10% in the test set.

After the division of vectors we implemented six supervised classifiers and analyzed the results. Naïve Bayes(NB) classifiers [22] are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naïve) independence assumptions between the features. We have applied three types of Naïve Bayes [15] classifiers on the data. They are mentioned below. Gaussian NB supports continuous valued features and models each as conforming to a Gaussian (normal) distribution.

$$p(x = v|C_k) = \frac{1}{\sqrt{2\prod\sigma_k^2}} e^{-\frac{(v-\mu_k)^2}{2\sigma_k^2}}. \tag{4}$$

where μ_k is the mean of the values in x associated with class C_k, σ_k^2 be the variance of the values in x associated with class C_k and $p(x = v|C_k)$ is the probability distribution of v given a class C_k. Multinomial NB [10] estimates the conditional probability of a particular word given a class as the relative frequency of term t in documents belonging to a class. The variation takes into account the number of occurrences of term t in training documents from that class including multiple occurrences.

$$p(x|C_k) = \frac{\left(\sum_i x_i\right)!}{\prod_i x_i!} \prod_i p_{k_i}^{x_i}. \tag{5}$$

where, x is the feature vector, p_{ki} is the probability of class C_k generating the term x_i. Bernoulli NB generates boolean value/indicator about each term of the vocabulary equal to 1 if the term belongs to examining document, if not it marks 0. Non-occurring terms in document are takes into document and they are factored when computing the conditional probabilities and thus the absence of terms is taken into account.

$$p(x|C_k) = \prod_{i=1}^{n} p_{k_i}^{x_i}(1 - p_{ki})^{1-x_i}. \tag{6}$$

where p_{ki} is the probability of class C_k generating the term x_i. k-NN [9] algorithm is a non-parametric method used for classification. The input consists of the k closest training examples in the feature space.

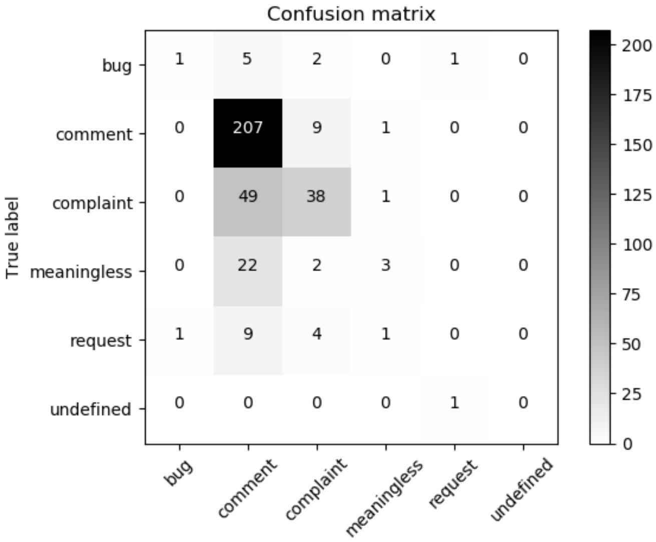


Fig. 1. Random Forest Confusion Matrix

The test sample is classified into a particular class as an output, depending upon the majority of the classes of its k-nearest neighbours. In plain words, if you are similar to your neighbours, then you are one of them. After sufficient experimenting, the value of k equals to 4 was taken. Binary SVM can be converted into a multiclass classifier using standard one versus one and one versus all.

We have used one versus all technique, in which a k-class problem is viewed as k many 2-class problem. In the training process, k binary classifiers are trained and each classifier tries to separate itself from $(k - 1)$ classes.Random forests [2] operate by constructing a number of decision trees at training time and outputting the class that is the mode of the classes. After some trial-and-error and close examination, the maximum depth as 200 and random state as 2 was taken to employ this classifier.

5 Result and Observations

After the experiments, an analysis of the six classifiers was done for the baseline by calculating some parameters, namely accuracy, precision score, recall score and F1 score with the help of confusion matrix. A confusion matrix [19], also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one.

Each column of the matrix represents the instances in a predicted class while each row represents the instances in an actual class (or vice versa). We constructed a confusion matrix for result analysis of our best performing Random Forest model. In our case, we put the instances of predicted classes in columns and instances of the actual classes in rows.

Hence a particular element of the matrix, say $CM[i][j]$ represents the number of feedback which were of class i but predicted as j. So when $i = j$, that is, the diagonal elements refer to the number of correctly predicted documents. The confusion matrix that we obtained is shown below. From the confusion matrix displayed in Figure 1, we infer that the maximum number of errors were found in differentiating between 'comment' and 'complaint' classes.

This was followed by the errors found in differentiating between 'meaningless' and 'comment'. Most correct predictions were from 'comment' class. The degree to which the result of a measurement, calculation or specification conforms to the correct value or a standard is called accuracy. It is the ratio of total number of correctly predicted documents to total number of documents.

$$\text{Accuracy} = \frac{\sum\limits_{i=1}^{6} \text{CM}[i][i]}{\sum\limits_{i=1}^{6}\sum\limits_{j=1}^{6} \text{CM}[i][j]} \times 100. \tag{7}$$

where CM = Confusion Matrix. Precision value for a class is the ratio of related information out of retrieved information to total retrieved information. Here we have taken average precision value of all classes.

$$\text{Precision} = \frac{1}{6}\sum_{i=1}^{6} \frac{\text{CM}[i][i]}{\sum\limits_{j=1}^{6} \text{CM}[i][j]}. \tag{8}$$

Recall value for a class is the ratio of related information out of retrieved information to total related information. Here we have taken average recall value of all classes.

$$\text{Recall} = \frac{1}{6}\sum_{i=1}^{6} \frac{\text{CM}[i][i]}{\sum\limits_{j=1}^{6} \text{CM}[j][i]}. \tag{9}$$

The F_1 score can be interpreted as a weighted harmonic mean of the precision and recall,where an F-beta score reaches its best value at 1 and worst score at 0.

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{10}$$

6 Error Analysis

After conducting the aforesaid experiments, we found a few reasons for the occurrence of errors:

- Most of the documents were classified into a single class, but some of them (about 4.5 percent) were classified into more than one class, e.g "Its nothing but it consumes a large amount of CPU and memory" was assigned both 'bug' and 'comment' classes. This creates an ambiguity for the classifier during training.

Table 3. Observations.

Classifier	Accuracy	Precision	Recall	F1 Score
Gaussian NB	53.42	0.53	0.53	0.53
Multinomial NB	55.59	0.56	0.56	0.56
Bernoulli NB	55.09	0.55	0.55	0.55
SVM	58.49	0.59	0.59	0.59
k-NN	57.65	0.58	0.58	0.58
Random forest	69.74	0.68	0.68	0.68

- The dataset was highly imbalanced; ‘bug’ and ‘undefined’ classes have 92 and 25 feedback respectively. On the other hand, ‘comment’ and ‘complaint’ classes have 1096 and 2034 feedback respectively.
- Due to the uneven distribution of data, a couple of classes have very few documents. This affects the dataset division process (into train and test set) as those few documents might end up at either train set or test set. This results in ramifications.

7 Conclusion and Future Scope

Working on multiclass classification that too for six classification of unbalanced dataset is not an easy task in the field of natural language processing & machine learning. After preprocessing of corpus, we employed 10-fold cross-validation method for training and testing purpose. We employed various machine learning algorithms to get the best model. Initially, we achieved an accuracy of 53.42% using Gaussian Naïve Bayes algorithm. Finally we got an accuracy of 69.74% using Random Forest, followed by accuracy of 55.59% using Multinomial Naïve Bayes, 55.09% using Bernoulli Naïve Bayes, 58.49% using SVM and 57.65% using k-NN classifiers respectively. Seeing the advancement in sentiment and text classification by deep learning [7, 23], in future we wish to explore deep learning for better accurate model.

Acknowledgments. Thanks to the IJCNLP 2017 shared task Customer Feedback Analysis organizers who have made their datasets available and thanks also to an anonymous reviewer for extensive and useful comments.

References

1. Bentley, M., Batra, S.: Giving voice to office customers: Best practices in how office handles verbatim text feedback. In: Big Data (Big Data), 2016 IEEE International Conference on. pp. 3826–3832. IEEE (2016)
2. Breiman, L.: Random forests. Machine learning , vol. 45, no. 1, pp. 5–32 (2001)
3. Chakankar, A., Mathur, S. P., Venuturimilli, K.: Sentiment analysis of users’ reviews and comments (2012)
4. Chaudhary, A., Kolhe, S., Kamal, R.: An improved random forest classifier for multi-class classification. Information Processing in Agriculture , vol. 3, no. 4, pp. 215–222 (2016)

5. Fang, X., Zhan, J.: Sentiment analysis using product review data. Journal of Big Data , vol. 2, no. 1, p. 5 (2015)
6. Gamon, M.: Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In: Proceedings of the 20th international conference on Computational Linguistics. p. 841. Association for Computational Linguistics (2004)
7. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for large-scale sentiment classification: A deep learning approach. In: Proceedings of the 28th international conference on machine learning (ICML-11). pp. 513–520 (2011)
8. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 168–177. ACM (2004)
9. Kataria, A., Singh, M.: A review of data classification using k-nearest neighbour algorithm. International Journal of Emerging Technology and Advanced Engineering , vol. 3, no. 6, pp. 354–360 (2013)
10. Kibriya, A. M., Frank, E., Pfahringer, B., Holmes, G.: Multinomial naive bayes for text categorization revisited. In: Australasian Joint Conference on Artificial Intelligence. pp. 488–499. Springer (2004)
11. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: IJCAI. , vol. 14, pp. 1137–1145. Montreal, Canada (1995)
12. Li, H., Jiao, R., Fan, J.: Precision of multi-class classification methods for support vector machines. In: Signal Processing, 2008. ICSP 2008. 9th International Conference on. pp. 1516–1519. IEEE (2008)
13. Liu, C.-H., Moriya, Y., Poncelas, A., Groves, D.: Ijcnlp-2017 task 4: Customer feedback analysis. Proceedings of the IJCNLP 2017, Shared Tasks pp. 26–33 (2017)
14. Liu, G., Zhang, X., Zhou, S.: Multi-class classification of support vector machines based on double binary tree. In: Natural Computation, 2008. ICNC'08. Fourth International Conference on. , vol. 2, pp. 102–105. IEEE (2008)
15. McCallum, A., Nigam, K., et al.: A comparison of event models for naive bayes text classification. In: AAAI-98 workshop on learning for text categorization. , vol. 752, pp. 41–48. Citeseer (1998)
16. Mukherjee, S., Bhattacharyya, P.: Feature specific sentiment analysis for product reviews. In: International Conference on Intelligent Text Processing and Computational Linguistics. pp. 475–487. Springer (2012)
17. Pal, M.: Random forest classifier for remote sensing classification. International Journal of Remote Sensing , vol. 26, no. 1, pp. 217–222 (2005)
18. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. pp. 79–86. Association for Computational Linguistics (2002)
19. Townsend, J. T.: Theoretical analysis of an alphabetic confusion matrix. Perception & Psychophysics , vol. 9, no. 1, pp. 40–50 (1971)
20. Yin, D., Hu, Y., Tang, J., Daly, T., Zhou, M., Ouyang, H., Chen, J., Kang, C., Deng, H., Nobata, C., et al.: Ranking relevance in yahoo search. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 323–332. ACM (2016)
21. Yuan, P., Chen, Y., Jin, H., Huang, L.: Msvm-knn: Combining svm and k-nn for multi-class text classification. In: Semantic Computing and Systems, 2008. WSCS'08. IEEE International Workshop on. pp. 133–140. IEEE (2008)
22. Zhang, H.: The optimality of naive bayes. AA , vol. 1, no. 2, p. 3 (2004)

23. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. In: Advances in neural information processing systems. pp. 649–657 (2015)

ISSN 1870-4069

Opinion Summarization Using Product Feature Based Review Classification

K Vimal Kumar[1], Pawan Upadhyay[1], Abheet Kumar Gupta[2],
Deepanshu Wadhwa[3]

[1]Jaypee Institute of Information Technology,
India

[2]Valuefirst Digital Media Private Limited.,
India

[3]Morgan Stanley,
India

vimal.kumar@jiit.ac.in, pawan.upadhyay@jiit.ac.in,
{abheet5080, deepanshu1404}@gmail.com

Abstract. As the e-commerce websites are expanding on a large scale, the number of reviews of products are increasing at a very rapid pace along with it. The e-commerce websites as well as manufacturers ask their customers explicitly to review their product. It helps both the fellow customers and the manufacturers to get a better knowledge about the product along with the related services that they are going to offer/get. So, with the increasing count of reviews, it becomes difficult for a user searching for a particular product to analyze all the reviews of that product in a very short period of time and make an informed decision on whether to purchase the product or not. In this research paper, we aim to mine and provide a sentiment analysis of the reviews received for a product. The proposed work is different from others as the aim of the system is to provide a sentiment insight of the features of the product rather than the complete review/sentence and visualize it properly to provide an easy way for the customers to make a proper decision. Our approach is based on tools and techniques from the fields such as opinion mining, natural language processing, and sentiment analysis.

Keywords. Opinion mining, sentiment classification, part of speech tagging, opinion summarization, reviews

1 Introduction

With the rapid increase in the e-commerce and the number of products sold online, the number of reviews for these products keeps increasing at an exponential rate. These reviews act as a good source of information both for the manufacturers and the customers. However, the millions of data obtained in the form of reviews make it difficult for the customers to segregate the product reviews on the basis of their polarity

pp. 73–83; rec. 2018-01-22; acc. 2018-03-05

(i.e. Positivity or Negativity) and to make a wise decision while buying the product. The overall rating of the product gives the customer an insight of the overall view of the product but not about its features. The customer still has to go through the large quantity of reviews to get feature-based polarity. The proposed method has been introduced to address these issues in a very efficient way. It deals with:

a) Extracting the polarity of the product reviews on the review level as well as feature level.
b) Summarization of the product features on the basis of their polarity.

Unlike the existing approaches that classify the reviews on the basis of polarity obtained on the product level, the proposed approach classifies the reviews on the feature level and not just on the sentence level. The current approaches extract the most common phrase/noun existing in the review to find the features of the product.

The proposed approach differs in the way that it extracts the features from the most popular e-commerce websites and use them as features. If any of the features or the most similar word to that feature appears in the review, they are used to find the opinion of users about that feature.

2 Related Work

The proposed work is closely related to the work done by Minqing Hu et al. [2], DuyKhang Ly et al. [3] on product review summarization from a deeper perspective. The existing method uses POS Tagger to identify the part-of-speech and further uses association rule mining to identify frequent explicit product features.

It then performs usefulness pruning on it to remove the useless words from the corpus. The method then identifies the sentences based on their positive and negative score. Based on the positive and negative score, the system summarizes the complete review. Proposed work is different from the existing in three ways:

i) The existing approach uses POS tagging, association rule mining, usefulness pruning to identify the product facets out of a given review related to the product. On the contrary, the proposed method identifies the frequent features of the product by scraping the features from various leading e-commerce websites. Furthermore, the features are added to the feature dictionary of the product based on the category in which the product lies. This saves a lot of processing time in the later processing phases.

ii) Apart from the technique that the existing method is using to score the opinion words, the system uses a well-defined corpus to identify feelings from the opinion word as happy, sad, arrogant, or excited. Thus, each of the opinion word will be scored not just on the basis of positivity or negativity that the word possesses, but according to various feelings that human beings possess.

iii) The opinion words are extracted from the reviews not just by identifying the nearest adjective word as proposed in the existing algorithm, but the proposed method introduces bigram and trigram approach to find the opinion word. This method first assumes the opinion word to exist as a bigram, then as a trigram

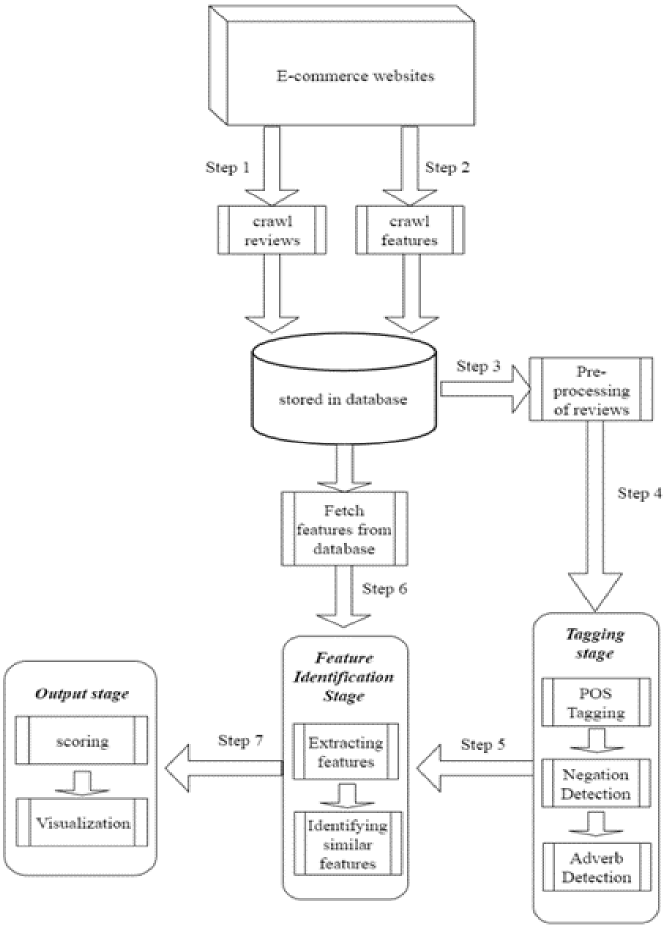


Fig. 1. The proposed architecture.

and so on up to n-gram. The opinion word is thus found and used to score the feature accordingly.

The existing work identifies features that appear explicitly as noun or noun phrase. For example, “The lens is awesome”. Here lens is the direct feature obtained. In this proposed system, both the explicit feature and implicit features are identified and in the implicit feature identification, most semantically similar features are found from the predefined feature list using cosine similarity between the vectors.

3 Proposed Technique

Figure 1 shows the detailed architecture of the proposed system. The method considers product names as input and gives its detailed summary of each feature of the product

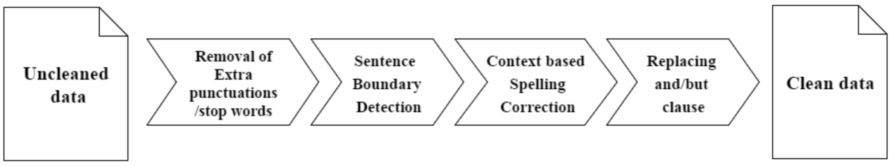
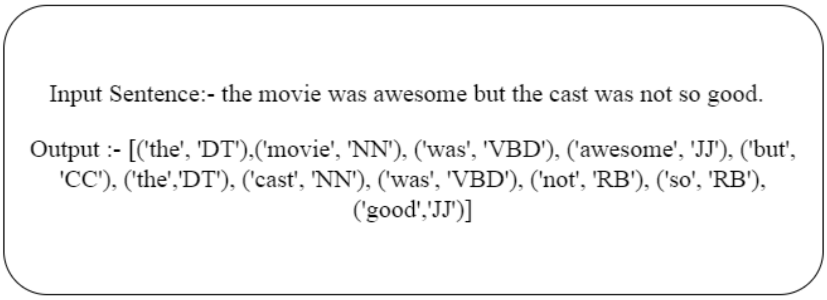

Uncleaned data
Removal of Extra punctuations /stop words
Sentence Boundary Detection
Context based Spelling Correction
Replacing and/but clause
Clean data

Fig. 1. Preprocessing steps.



Input Sentence:- the movie was awesome but the cast was not so good.

Output :- [('the', 'DT'),('movie', 'NN'), ('was', 'VBD'), ('awesome', 'JJ'), ('but', 'CC'), ('the','DT'), ('cast', 'NN'), ('was', 'VBD'), ('not', 'RB'), ('so', 'RB'), ('good','JJ')]

Fig. 3. Example of POS Tagging.

as output along with its polarity. The system performs the opinion generation of reviews as well as the features of the product in mainly five steps.

3.1 Review and Feature Extraction

The reviews of the product searched are extracted from different e-commerce websites along with their key features. This step gives us the advantage over the method proposed by Minqing Hu et al [2], D. Ly et al [3], in which the method does not identify the frequent feature dataset for the product and thus optimizing/reducing the steps involved in the process.

The step also involves general feature addition common to all products across the product category to the above feature dataset extracted from the websites. The extracted features as well as reviews are stored in the database which enables us to reduce the re-extraction of features every time a product is to be summarized.

3.2 Preprocessing Stage

Preprocessing stage is shown completely in figure 2. It consists of mainly four steps to clean the data. To describe about the preprocessing stage, consider the input sentence as:

“*The product has amazng features but its size is toooo big…!!!!!!*”

The reviews are taken and fed into first block which removes the extra punctuations and stop words out of it.

Output: *Product has amazng features but its size is toooo big.*

The second step involves sentence boundary detection in which different sentences in the same reviews are merged as a list.

Output: *Product has amazng features but its size is toooo big.*

Next step involves spelling/slang correction to correct any incorrect spellings and slangs used in the reviews.

Output: *Product has amazing features but its size is too big.*

Final step involves but/and clause removal and breaks a complex sentence into two individual sentences.

Output: *Product has amazing features. Its size is too big.*

3.3 Review Tagging

After the preprocessing of reviews, the cleaned reviews are fed to the tagging stage. The tagging stage involves generation of a dictionary (*key-value pair*) which has the key as identified tokens of the reviews and values as a list of properties related to a token. The three properties are:

- Part of Speech Tagging,
- Negation Detection,
- Adverb Detection.

3.3.1 Part of Speech Tagging

Part of speech tagging refers to tagging each token of the sentence according to the part of speech represented by it (whether the token is a noun, a verb, a determiner, etc.). It is required so as to remove the insignificant part of speech as well as in the later feature extraction phase. It forms the list in the form of tuples containing individual tokens and their corresponding part of speech tag. Example is as shown in figure 3.

The extracted POS tagged data is further used in the next processing stage. The data is saved in the database in the form of dictionary containing each token of a sentence as a key and the value as a list containing POS tag along with other factors mentioned later.

3.3.2 Negation Detection

It is very important to detect the negation in the sentence because it can reverse the polarity of the adjective (used to identify the polarity of the sentence). Generally, the negating word is present before the opinion describing word (i.e. adjective), therefore the process involves the detection of negating word with a help of predefined list of negating words and adding a flag "neg y" to all the tokens value list till the end of the sentence. Else a "neg_n" flag is appended to the list to indicate that negating word is not present up to that token in the review. A list of negation words is maintained in advance and the tokens are compared with the list to find if the negation words are present or not.

3.3.3 Adverb Detection

The adverb's present before an adjective can have a very good impact on the sentiment expressed by a sentence. Adverbs can drastically increase the sentiment score as well as decrease it. The adverbs are extracted from the sentence through the part-of-speech tagged data and will be checked whether the adverb is an adverb of degree or not. If yes, that particular adverb is taken and checked further whether the adverb is a strong intensifying adverb or a weak intensifying adverb. For example, ‘Very good’ in a sentence will have a better score as compared to ‘good’. So, adverb detection becomes a mandatory part of the process.

3.4 Feature Identification

This step involves identifying the key-category of features of the product on which the people have expressed their opinion. Before discussing the methods involved in identifying the features, let us first look at the kinds of examples that'll be handling. The aim of the proposed system is to find what feature of the product is liked or disliked by the user. It is a very important and crucial step to identify the feature about which the people are expressing their views. Let us discuss an example of a digital camera:

“*The lens of this digital camera is awesome.*”

In the above sentence, it is visible that a reviewer is satisfied with the lens of the digital camera; lens is the key-feature in this review that the reviewer has mentioned about. Whereas it can be observed from the sentence that the feature of the product is explicitly mentioned, there are some reviews where the features are hidden or implicit and hard to find. For example,

“The camera doesn't fit in the pocket.”

The challenge in the above example is to identify the key-feature (i.e., size or dimension) of the camera about which the user has expressed his views. Thus, the proposed system deploys finding key-category of the feature mentioned in the review by the reviewer. Considering the above example, the key-category of the feature mentioned is dimension. This has an advantage that when it needs to summarize a key-category of feature that can include all those features which lie under a particular key-category of feature. For example, [‘size’, ‘width’, 'height’] all will lie under a key-category of dimension and thus saving us from re-computing the reviews which lie under a particular key-category of features in the later summarization stage.

The list of key-category of features is obtained from the same e-commerce websites from which the product reviews are obtained. This helps us in saving the process of defining the key-category of features. The basic idea is to extract all the nouns or noun phrases in the reviews and find the semantically similar key-category of feature from the obtained list, which is performed in the Word embedding stage that follows

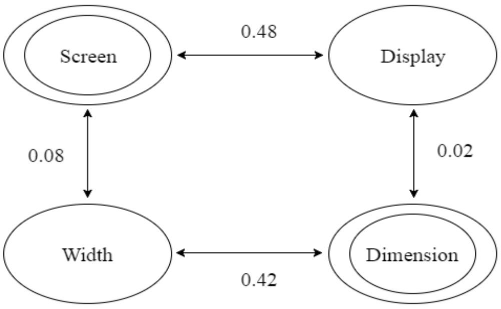
3.4.1 Word Embedding Stage

The key-category of semantically similar features are identified in this stage. Word2vec creates vectors that are distributed numerical representations of word features, such as the context of individual words. The main advantage of Word2vec is that it performs the grouping of those vectors that are of similar words in vector space.

That is, it detects similarities mathematically. The method employed in the word2vec stage to obtain the key-category, involves finding the cosine similarity between the vector of keyword to be categorized and one by one vector of elements present in key-category list.

This similarity would result in a similarity score and if the similarity score is greater than a threshold value (in our case it was 0.30) and maximum score among all key-categories, then the review will be categorized to a particular key-category of features.

The nouns or noun phrases which are used to categorize the review to a particular key-category of features, are saved for identification of opinion words in the next step (III.E).



Here, dimension and screen are identified as key category of features of the product. When display and width come as new features, they are first check for similarity with existing ones. Display is found to be most similar to screen and width is found to be most similar to dimension. Thus, dimension and width, display and screen are merged together.

Fig. 4. Similar feature identification.

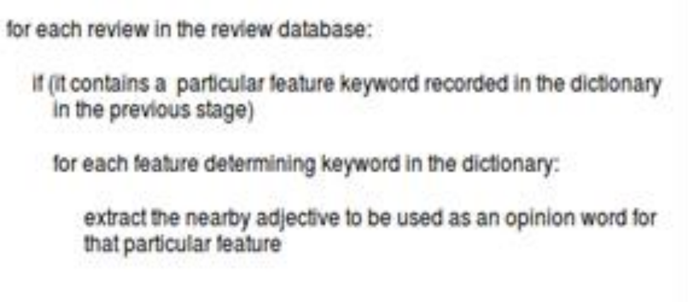
3.5 Opinion word extraction and summarization

The next task is to identify the opinion words. These opinion words give a clear view of what a particular person thinks about the product as well as about the particular feature of the product.

The method involves the identification of adjectives which will be used as the opinion word in this proposed method. For each key-category of feature, its opinion word is extracted and saved for computation purpose in further stages. Let's first see how the opinion words are extracted (as shown in Figure 5).

Let's understand the above algorithm with the help of an example:

The image quality of this camera is poor.



```
for each review in the review database:
  if (it contains a  particular feature keyword recorded in the dictionary
    in the previous stage)
    for each feature determining keyword in the dictionary:
      extract the nearby adjective to be used as an opinion word for
      that particular feature
```

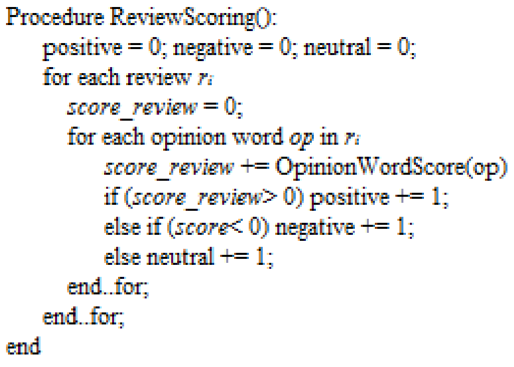
Fig. 5. Steps for opinion mining.

Here if feature extraction is applied on the above example, a dictionary will be created which will contain the specific feature-word present in the review (e.g. {review1: [quality]}). This feature-keyword from the dictionary will be used in this stage to extract the nearby opinion word (adjective) which in above example is poor.

3.5.1 Score Calculation and Visualization

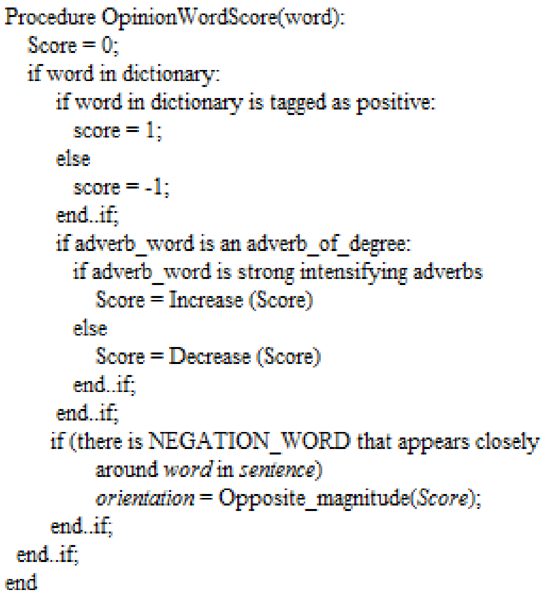
The opinion word obtained in the previous stage are saved for each review and for each feature, which helps in giving the opinion summary at sentence level as well as at product feature level.

A dictionary of all the words tagged as positive and negative is taken and for each opinion word in the review, the opinion word is looked up in the dictionary for classifying the word as positive or negative, negation detection is done to identify the exact polarity of the opinion word (negation can revert the polarity of the opinion word) and finally adverb detection is applied which can increase the score of a particular opinion word, if present. The pseudo code mentioned below shows how the scoring is done:



```
Procedure ReviewScoring():
    positive = 0; negative = 0; neutral = 0;
    for each review $r_i$
      score_review = 0;
      for each opinion word op in $r_i$
          score_review += OpinionWordScore(op)
          if (score_review> 0) positive += 1;
          else if (score< 0) negative += 1;
          else neutral += 1;
      end..for;
    end..for;
end
```

For feature level Opinion identification, the opinion word obtained for word is passed into OpinionWordScore (opinion_word) function and the obtained score is used to identify the polarity of the review.



```
Procedure OpinionWordScore(word):
   Score = 0;
   if word in dictionary:
       if word in dictionary is tagged as positive:
         score = 1;
       else
         score = -1;
       end..if;
       if adverb_word is an adverb_of_degree:
         if adverb_word is strong intensifying adverbs
            Score = Increase (Score)
         else
            Score = Decrease (Score)
         end..if;
       end..if;
      if (there is NEGATION_WORD that appears closely
            around word in sentence)
            orientation = Opposite_magnitude(Score);
      end..if;
 end..if;
end
```

The next step is opinion summarization which involves the summarization of overall user review's opinion as well as feature level product opinion. The score from opinion word about a particular feature present in the review is calculated as either positive, negative or neutral. Percentage of each is calculated and is used to visualize the data in form of pie chart or gauge chart. The frequency (occurrence) of all the features in reviews is calculated and is used to rank the features. The feature with a greater number of reviews is ranked higher and is given more preference as compared to others. This is done to throw light on that feature which is highly talked about by the customers.

Table 1. Categories used for testing.

	Tagged as positive	Tagged as negative
Predicted as positive	TP	FP
Predicted as negative	FN	TN

4 Result Evaluation

The proposed system has been implemented in python language through the use of various Natural Language Processing techniques. This system is evaluated by crawling reviews from different leading e-commerce websites. Next, these reviews are crowd sourced to find the opinion of people towards various features present in the review. Then, the results were compared to find the accuracy of the proposed method. The main difficulty in the review opinion generation is the presence of sarcasm in the reviews. The reviews can be subjective and the satire in the reviews make them difficult to judge whether a review is positive or negative. The experimental results are shown in the Table 1 & 2 in the form of accuracy, precision, and recall.

Table 2. Result of the proposed system.

Accuracy	81.5%
Precision	0.8558
Recall	0.8079

5 Conclusion and Future Work

In this research paper, the proposed method tends to obtain the features in a review and the opinion of people towards those features. The main objective is to get the viewpoints of people for the products sold online and generate a report of the features along with opinions of people towards them. This is beneficial both for the customers and the manufacturers selling their products online. The results of the experiments suggest that the algorithm for the proposed method works quite efficiently.

Our system finds the features of the products stored as nouns and merges similar features to make them single. The future work plans to identify the solutions to the problems yet not covered in the proposed work. Sentences with sarcasm will be checked for the identification of judgment of people towards the product. Furthermore, strength of adverbs is not identified efficiently in the proposed method. We also plan to improve the efficiency for finding the key-category of features from the reviews containing the implicit features.

References

1. Mishra, N., Jha, C.K.: Classification of Opinion Mining Techniques, International Journal of Computer Applications, vol. 56, no. 13 (2012), pp 1–6. DOI:10.5120/8948-3122
2. Hu, M., Liu, B.: Mining and Summarizing Customer Reviews. In: Proceedings of the ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD04), pp. 168–177 (2004)
3. Ly, D., Sugiyama, K., Lin, Z., Kan, M.: Product review summarization from a deeper perspective. In: Proceedings of the Joint Conference on Digital Libraries, pp. 311–314 (2011)
4. Kokkoras, F., Lampridou, E., Ntonas, K., Vlahavas, I.: Summarization of multiple, metadata rich, product reviews. In: Workshop on Mining Social Data (MSoDa), 18th European Conference on Artificial Intelligence (ECAI '08), Patras, Greece (2008)
5. Mugdha, M., Bharat T.: A framework for summarization of online opinion using weighting scheme. In: Proceedings of the Pervasive Computing (ICPC). International Conference, pp 1–6 (2015)
6. Krishna, G.R., Jothi S., Minojini, N., Sowmiyaa, P.: Survey of Various Opinion Mining Approaches, in Proceedings of the International Journal of Innovative Research in Computerand Communication Engineering. vol. 3, no. 3 (2015)
7. Kumar, P., Mohd, H.: Analytical study of Feature Extraction Techniques in Opinion Mining. In: International Conference on Advance in Computing and Information Technology, page 85–94 (2013)
8. K. Ly, K. Sugiyama, Z. Lin, M. Kan.: Product review summarization based on facet identification and sentence clustering. CoRR'11

9. Vinodhini, G., Chandrasekaran, RM.: Sentiment Analysis and Opinion Mining: A Survey. International Journal of Advanced Research in Computer Science and Software Engineering, vol. 2, no. 6 (2012)
10. Brisson, L., Jean-Claude, T.: Opinion mining on experience feedback: A case study on smarphones reviews, Conference: RCIS 2015. In: IEEE 9th International Conference on Research Challenges in Information Science (2015). doi: 10.1109/RCIS.2015.7128879
11. Dalal, M.K., Zaveri, M.A.: Semi supervised learning bases opinion summarization and classification for online product reviews. Applies Computational Intelligence and Soft Computing, vol. 2013 (2013)

ISSN 1870-4069

Translation of Wikipedia Category Names by Pattern Analysis

Thang Ta Hoang[1,2], Alexander Gelbukh[1]

[1] Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Mexico

[2] Dalat University,
Vietnam

tahoangthang@gmail.com, gelbukh@cic.ipn.mx,
thangth@dlu.edu.vn

Abstract. Category names, which comply to the naming conventions, have been developed by the editor community of Wikipedia for years. They can be divided into category patterns that serve translation tasks and other research works in the NLP fields. In this paper, we propose a translation model based on pattern analysis and build a translation tool, which can semi-automatically translate over 7000 new categories from English to Vietnamese with a highly reliable outcome. The result has a huge role in contributing new category names for Wikipedia and reduces the repetitive and tedious edits of editors.

Keywords: Wikipedia category, category translation, naming pattern analysis.

1 Introduction

It is not regular to see a successful online project which obtains massive human cooperation from all around the world, without discriminating who they are or where they come from. Wikipedia has been proved that humans together be able to develop such tremendous content with 57,578,491 articles in 323 language projects[3] which offer their precious data, including the category taxonomy to the community of researchers. Wikipedia editors, based on their own understanding, contribute manually new categories and category classification or reuse these data from other language projects, particularly English Wikipedia. We trigger an idea that to build a tool for translating these categories to overcome this tedious task.

Editors thus can focus more on devoting other helpful information to Wikipedia. Category names can be basically viewed as noun phrases with a terse, explicit, and descriptive meaning. By glancing at a random category name, readers and editors can recognize what it is talking about and which articles will belong to it. For instance, "`Category:French scientists`" consists of articles about scientists who hold a French passport.

[3] https://meta.wikimedia.org/wiki/List_of_Wikipedias

Instead of using traditional translation methods or deep learning approaches, we use naming pattern analysis for an online multilingual translation approach from combining with Wikidata, other sister projects of Wikipedia to see how the two projects can work together to generate the best results.

In this way, if we can make the pattern alignments between languages, we can proceed to translate from a certain language to any language without using complex algorithms and pre-defined bulky databases. In this paper, Section 2 contains relevant works with category translation and taxonomy, then we present our translation method in Section 3. Section 4 is about the translation tool and how it works. Finally, we execute the experiment and evaluate the results in Section 5; summarize the paper and future works in Section 6.

2 Related Works

Wikipedia category taxonomy is considered a free, enormous, and valuable resource as well as a research object appearing in a lot of papers. There are some works about extracting or deducing semantic relations from the alignment and comparison of category entities. The category graph was constructed by a graph-theoretic analysis to indicate its ability to handle NLP tasks. Zesch and Gurevych utilized the multilingual power of Wikipedia in applying NLP algorithms for languages instead of self-built semantic WordNet [18].

The authors transferred semantic relatedness algorithms defined on WordNet to the Wikipedia category graph and evaluated its coverage and the performance of these algorithms. Chernov et al. extracted the semantic information from links between categories on Wikipedia [2]. They concluded that the outcome was useful for forming a Wikipedia semantic schema to broaden search capabilities and provide meaningful suggestions of editors when they contribute to Wikipedia articles.

Outside Wikipedia, Pasupat and Liang applied queries and a new zero shot learning task to extract category entities from web pages containing semi-structured data [11]. Focus on name labels and their construction is also a research aspect of Wikipedia categories. Bøhn and Nørvag chose categories from three areas: (1) people, (2) organizations, and (3) companies for improving the NE recognition.

The authors presented some wildcard patterns which did not clarify about category types they belonged to and their semantic relationships [1]. Ponzetto and Strube worked on `isa` and `notisa` patterns to derive numerous semantic relations from the category system and compared the result quality with ResearchCyc [12]. Nastase and Strube decoded category names by arranging them into various category types and patterns in order to simplify and reproduce the relations between articles and categories [10].

These patterns, including two variables (X and Y) with their relationship, were analyzed from English names. We prefer to apply this classification to other languages and make alignments between languages for the translation task. Wang et al. inherited this classification to apply for a weakly supervised learning framework to collect relations from Chinese Wikipedia categories [17]. Generally, category names can be seen as noun phrases which we can apply a Statistical Phrase-Based method [6,7] to the translate process.

Based on this research, Pu et al. improved noun phrase translation of polysemous nouns (XY compounds) from Chinese and German [13]. Another method of noun phrase translation is to use Word-Based Model [8,9], which has more precise compared with previous adjacent methods and syntax-based methods [8]. As declared in Introduction Section, in this research, we try to use a light method that depends on an available multilingual source of Wikidata to execute the translation task.

3 Methodology

3.1 Pattern Alignment

In order to begin the translation process, we need to create pattern alignments between a source language and a destination language. We apply English and Vietnamese as two languages mentioned and translate category names from English to Vietnamese. Also, if we obtain the pattern alignments in more languages, we will have more languages to be translated. The patterns are split into two categories, English pattern (Ep) and Vietnamese pattern (Vp). We collect Eps analyzed from works [10,12] and make the alignment table representing the correlation between Eps and Vps. Table 1 lists some alignment rules between Eps and Vps.

On Wikipedia, naming conventions are standards formed and developed by the editor community, everybody should comply to these standards when they create new categories or arrange category taxonomy. The most benefit of naming conventions is to keep category names in a homogeneous way. In English, we refer to naming conventions in the Category:Wikipedia naming conventions[4] which describe in detail for every single case. Similarly, these conventions can be found at Wikipedia:Thể loại[5] in Vietnamese.

Table 1 only shows several alignment rules extracted from the Vietnamese community agreement. Actually, there are more rules than that; however, when we would like to contribute new categories to Vietnamese Wikipedia, we have to respect the naming conventions, so we only list some consensus patterns. The extension of the English pattern in P2 may be "XYZ" when we will work with the noun phrases containing nouns. For the noun phrases contain adjectives, we do not take them into account. When translating to Vietnamese, this pattern will be reversed to "ZYX" but not always for all cases.

For example, we divide category "Computer science awards" into three parts: X = "computer", Y = "science" and Z = "awards". In Vietnamese, these parts are translated to X = "máy tính", Y = "khoa học", and Z = "giải thưởng" while the result is "Giải thưởng khoa học máy tính". In another example, the category "University College London" is cracked into X = University (Đại học), Y = College (Cao đẳng) and Z = London (Luân Đôn). If we apply the same pattern as the previous example, we will have wrong Vietnamese name "Luân Đôn Cao đẳng Đại học" while the correct name must be "Đại học Cao đẳng Luân Đôn".

[4] https://en.wikipedia.org/wiki/Category:Wikipedia_naming_conventions

[5] https://vi.wikipedia.org/wiki/Wikipedia:Thể_loại

Table 1. Some typical alignment rules between Eps and Vps.

ID	Ep	Vp		Examples
P1	X	X		en:Science,X = science vi:Khoa học, X = khoa học
P2	XY	YX X của Y		en:Computer science X = computer, Y = science vi:Khoa học máy tính X = máy tính, Y = khoa học en:Adele albums X = Adele, Y = albums vi:Album của Adele X = Adele, Y = album Adele is a person, so we use “của” meaning “of”

P3	X by Y	X theo Y		en: Cities by country
				X = cities, Y = country
				vi: Thành phố theo quốc gia
				X = thành phố, Y = quốc gia

P4	X in Y	X ở Y X tại Y XY		en: Cities in Vietnam X = Cities, Y = Vietnam vi: Thành phố Việt Nam vi: Thành phố ở Việt Nam vi: Thành phố tại Việt Nam X = thành phố, Y = Việt Nam
P5	X of Y	X của Y	XY	en:Birds of Vietnam X = Birds, Y = Vietnam vi: Chim Việt Nam X = Chim, Y = Việt Nam en:History of Mexico X = history, Y = Mexico vi:Lịch sử Mexico vi:Lịch sử của Mexico X = lịch sử, Y = Mexico

P6	X from Y	X từ Y		en:People from Hanoi
				X = people, Y = Hanoi
				vi:Người từ Hà Nội
				X = người, Y = Hà Nội

P7	X [VBN IN] Y	X được [VNB]Y	X do Y [VNB]	en:Films directed by Peter Lord X = films, Y = Peter Lord vi:Phim được đạo diễn bởi Peter Lord vi:Phim do Peter Lord đạo diễn X = phim, Y = Peter Lord
P8	X in Y (X is year)	Y năm X	YX	en:2018 in Vietnam X = 2018, Y = Việt Nam vi:Việt Nam năm 2018 vi:Việt Nam 2018 X = 2018, Y = Việt Nam

P9	X(s) in Y (X is year)	Y thập niên X		en:2010s in Vietnam
				X = 2010s, Y = Vietnam
				vi:Việt Nam thập niên 2010
				X = 2010, Y = Việt Nam

That's the reason why we have the manual evaluation step to be sure the accuracy for all translated category names. Some patterns (P2, P4, P5, and P6) fall into the ambiguous cases, which an English pattern can interpret into multiple Vietnamese patterns. To deal with this problem, we classify existing category names into featured groups in English and Vietnamese which meet some similar characteristics. We prioritize these groups by following orders: preposition matching, head matching [12] (prefix) and tail matching (suffix).

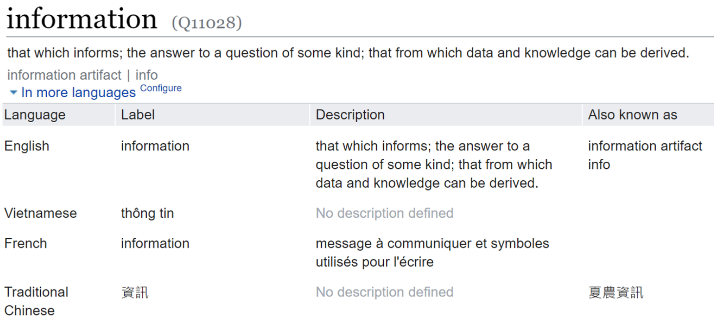
For example, group g1 includes category names with the head "Ca sĩ"(singer) may have these members, g1 = ["Ca sĩ Singapore", "Ca sĩ thế kỷ 20", "Ca sĩ Thái Lan"]. In another example, group g2 = ["Cities in Japan", "Radio in Mexico"] matches with preposition "in" while group g3 goes very well with the tail "Việt Nam" with elements such as g3 = ["Văn hóa Việt Nam", "Năng lượng ở Việt Nam", "Đập tại Việt Nam"]. In each group, we choose a category name candidate which has the highest matching point (M-point) with the translated category name by Dao's module [3].

We choose a category candidate in English and a category candidate in Vietnamese. Next, we calculate the mean of M-points between category candidates and category names in English and Vietnamese. Then, we compare this mean to a threshold (0.5). If the mean is higher or equal to the threshold, we choose it for the translation. Otherwise, we will drop this translation and move to the next one. English patterns (P2, P4, P5, and P6) can be written in the general form as X [prep] Y.

Some other prepositions (about, on, upon, over) categorized into this case have no agreement in Vietnamese Wikipedia, so we simply discard them. Pattern P7, X [VBN IN] Y is a complicated case that we have to take it out of our practice because English has a lot of irregular verbs, including their past and past participate forms. We will consider taking it in our next research to expand more translated results. Pattern P8, X năm Y is a naming convention that complies to Vietnamese Wikipedia agreements, but the pattern XY is still used in reality, so we create a redirection from pattern XY to pattern X năm Y.

Different from English, Vietnamese nouns do not have any notion of number or amount. For example, we can translate "cities" to "thành phố" and this name can be denoted for one city or many cities. Instead, to determine a noun in the plural or singular form in Vietnamese, we use plural markers (pluralizers) before it, such as những (several),các (several) and một (one). If we have no matter with pluralizers, we can remove them in translation to obtain a short category name while still guarantees an explicit meaning.

At last, we deal with long category names, which can be viewed as a combination of name components. For example, we take category "Information Technology in Mexico" as pattern "X in Y" when its components are X = "Information Technology" and Y = "Mexico". We continue to split the noun phrase "Information Technology" as pattern "XY" with X1 = "Information" and Y1 = "Technology". Our task now is to translate X1, Y1, Y into Vietnamese, and combine them to produce the result name "Công nghệ thông tin ở Mexico".



information (Q11028)

that which informs; the answer to a question of some kind; that from which data and knowledge can be derived.

information artifact | info

▾ In more languages Configure

Language	Label	Description	Also known as
English	information	that which informs; the answer to a question of some kind; that from which data and knowledge can be derived.	information artifact info
Vietnamese	thông tin	No description defined	
French	information	message à communiquer et symboles utilisés pour l'écrire	
Traditional Chinese	資訊	No description defined	夏農資訊

Fig. 1. Interwiki links of Item “Information” in Wikidata.

3.2 Wikidata as a Multilingual Semantic Source

An interwiki link (interlink) is responsible to connect two articles (or categories, templates, etc.) in two Wikipedia language editions together. This link helps readers and editors be able to find articles on various language editions for some common purposes such as content comparison, translation, broadening knowledge, or research. Interwiki links used to apply the classical structure when every language edition must store syntax lines (ex. `[[en:ArticleName]]` with `en` is English and `ArticleName` is an article name).

This triggers the problems of link maintenance and bulky data storage, Wikimedia organization launched Wikidata in October 2012 as a central data management platform for storing multilingual links and semantic relations [4,16]. Wikidata stores multilingual labels in two main types: Item (`Q`) and Property (`P`). Item and Property both include statements (semantic relations). Item is used to link article names, while Property works with properties.

Figure 1 indicates the multilingual labels of Item ”Information” (`Q11028`) in English, French, Vietnamese, and Traditional Chinese. From Wikidata, we can retrieve any term in any language from the English input. In this research, we use Wikidata as a source to search for Vietnamese terms instead of using bilingual dictionaries.

One of the important things that many researchers are concerned about Wikidata is its reliability [5]. Because of Wikidata’s policy, everyone can freely contribute to this project, so it is sometimes difficult to counter vandalism. Sarabadani et al. declared that vandalism edits were just 17 edits (0.17%) per 10000 samples [15] which pointed out that the vandalism is not considerable. In the translation model, we also run a manual evaluation; thus, we are confident to use the data of Wikidata.

3.3 Translation Model

We organize the translation order of patterns as following: X-pattern translation (single pattern which cannot be divided into smaller patterns), P-pattern translation (patterns with prepositions, i.e. `X [prep] Y` patterns), XY-pattern translation (noun phrase patterns) and O-pattern translation (other patterns). Figure 2 presents the model of how to translate category names.

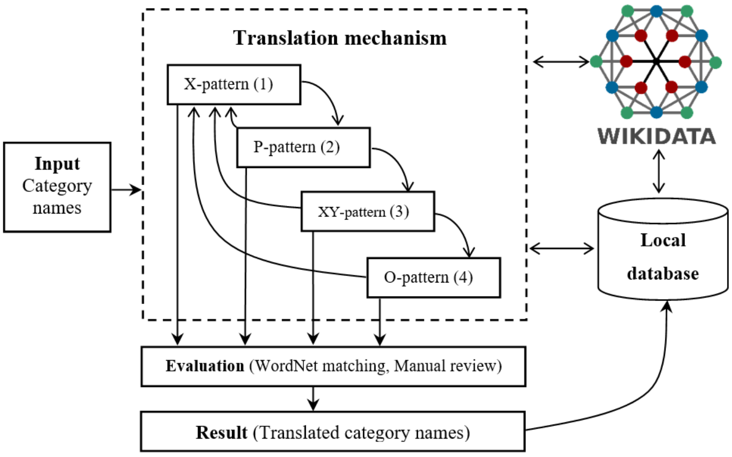
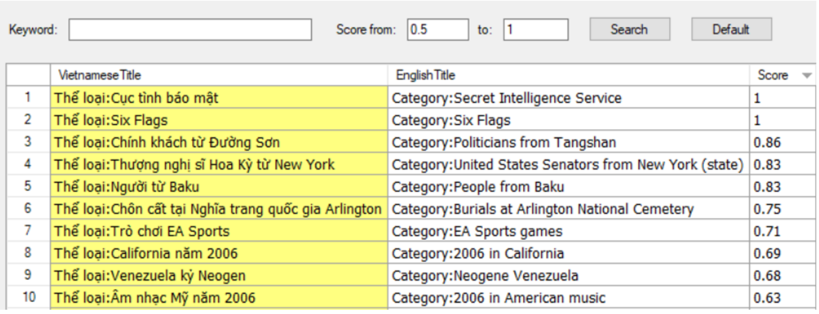


Fig. 2. The translation model.

Firstly, we collect English category names which do not have any corresponding name in Vietnamese or do not exist interlinks to Vietnamese Wikipedia on Wikidata as the input. In the translation mechanism in Figure 1, with category `A` in the list, we will proceed:

- `Step 1`: We treat `A` as X-pattern and check it at Wikidata. If we find Vietnamese name `B` respectively, we continue to check `B`. If `B` existed in Vietnamese Wikipedia and/or has no interwiki links, we stop translation and return a failure. Otherwise, we take `B` to the evaluation process. If not, we pass it to `Step 2`.
- `Step 2`: We check `A` to have prepositions or not. If yes, we split `A` into three parts: pre-preposition, preposition and post-preposition. We continue to repeat `Step 1` with these three parts and gather the results. If one of these parts do not have the corresponding Vietnamese name, we stop the translation. If not, we pass it to `Step 3`.
- `Step 3`: In this step, we will deal with noun phrase translation (XY-pattern). We begin by splitting `A` into two parts: the last word and the remains. Later, we repeat `Step 1` for these two. If one of these parts does not have a Vietnamese name when we check it at Wikidata, we continue to split `A` into two parts: last two words and the remains. We repeat `Step 1` for these two. We repeat until we have the two parts: the first word and the remains, but we do not still reach the results, it means a failure. If we can get the result parts, we reverse the parts (`XY -> YX`) then put in into the evaluation.
- `Step 4`: We apply this step to translate category names based on O-pattern which we find and integrate in the translation process when `Step 1`, `Step 2`, `Step 3` are not working. Depending on every category pattern, we will decide to pass `A` to `Step 1`, `Step 2`, or `Step 3`. The evaluation will be proceeded in the case we have the return results. Otherwise, the translation is ended.



Keyword: Score from: 0.5 to: 1

	VietnameseTitle	EnglishTitle	Score
1	Thể loại:Cục tình báo mật	Category:Secret Intelligence Service	1
2	Thể loại:Six Flags	Category:Six Flags	1
3	Thể loại:Chính khách từ Đường Sơn	Category:Politicians from Tangshan	0.86
4	Thể loại:Thượng nghị sĩ Hoa Kỳ từ New York	Category:United States Senators from New York (state)	0.83
5	Thể loại:Người từ Baku	Category:People from Baku	0.83
6	Thể loại:Chôn cất tại Nghĩa trang quốc gia Arlington	Category:Burials at Arlington National Cemetery	0.75
7	Thể loại:Trò chơi EA Sports	Category:EA Sports games	0.71
8	Thể loại:California năm 2006	Category:2006 in California	0.69
9	Thể loại:Venezuela kỷ Neogen	Category:Neogene Venezuela	0.68
10	Thể loại:Âm nhạc Mỹ năm 2006	Category:2006 in American music	0.63

Fig. 3. The screenshot of the result of translated category names.

The next step is to calculate the M-point of translated names with a candidate category name. We use Dao's module [3] to measure the similarity of category names, which estimates phrases by WordNet. If the M-point is equal or more than 0.5 (our pragmatic threshold), we keep this category name.

Eventually, we do a manual evaluation and store new category names into a local database. The extension job may be to create these new names in Vietnamese Wikipedia or upgrade the category taxonomy (RDF triples). For every translated category name, we also store its structure (Name-Analysis) that can be inherited to translate the next names.

4 Translation Tool

We built a tool so-called "`Wiki Category`" written by C# language on .NET Platform, which offers results for AutoWikiBrowser to import new categories to Vietnamese Wikipedia. We use a local database, including 58881 categories and 11569 articles (in both English and Vietnamese) as a buffer memory to reduce executed time for the translation process instead of retrieving data online from Wikidata APIs. The number of categories and articles will increase whenever the tool is active.

The tool works on some simple steps. First, it collects the article list (random articles, new recently added articles, articles by category, etc.), and untranslated category names in each article. Then, these data will be the input for the translation execution. Finally, a review process is responsible for the accuracy of translated categories (as in Figure 3) that humans can intervene to correct if needed.

For each category, we apply four labels for managing easily: `Not_create`, `Existing`, `Existing_Missing_Link` (no interwiki link) and `Redirected`. We will add controversial names to the blacklist to stop translating them in the future. Besides, the results can be exported to XML format for other uses.

5 Experiment

As of October 2021, we generated more than 7000 new categories in Vietnamese and contributed 6035 categories to Vietnamese Wikipedia. We randomly picked a sample

Table 2. The distribution of the number of pattern types in a random 1000 categories.

X-Pattern	P-Pattern	XY-Pattern	O-Pattern
0	796	181	118

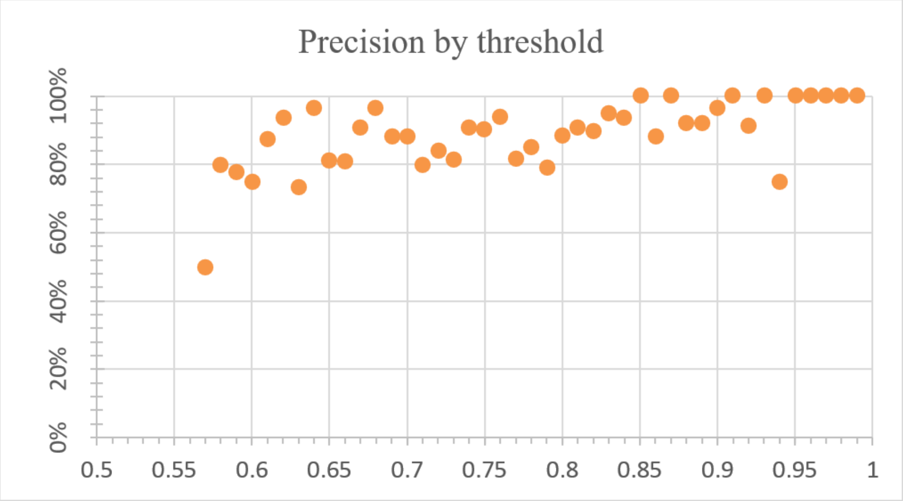


Fig. 4. The distribution of precision by threshold.

of 1000 random categories and counted the number of category names by pattern types, and then estimated the precision with a threshold of 0.5. In Table 2, P-pattern occupied the most number of category names, followed by XY-pattern and O-pattern. We realized that there were many category names belonging to more than one pattern type; however, we did not put them into the statistics. We did not surprise that the number of X-pattern categories was zero or a very few. This pattern type is quite simple (single and popular names) so editors probably translated almost of them to Vietnamese.

We manually scrutinized the correctness of each category name of the sample set. Subsequently, we received a precision of 0.89 on 1000 samples, which is likely high, but not our expectation. We reviewed the results and found that category names of XY-pattern, particularly long names, had the most error rate of 5.9% on all pattern types. If compute XY-pattern only, the error rate is 32%. This means our inverse rule (`XY -> YX` and the extension `XYZ -> ZYX`) can not work for all cases.

In Figure 4, the precision is relatively high and increases proportionally with M-point. We found that the precision is nearly 1 if the threshold is from 0.95. With M-point less than 0.7, the precision is 0.86; M-point from 0.7 and to less than 0.8, the precision is 0.85; M-point from 0.8 and to less than 0.9, the precision is 0.92; the precision is 0.94 with M-point more than 0.9.

6 Conclusion and Future Work

In this paper, we presented category patterns used to determine category name structure in English and Vietnamese. The translation model used several simple steps based on the combination of alignment rules and our evaluation method. In the experiment, we built `Wiki Category`, a tool which produced more than 7000 new categories for Vietnamese Wikipedia. We obtained high precision of 0.89 with the threshold of 0.5 on 1000 translated categories. Besides, we still had a problem with XY-pattern when it still contains many translation errors.

The outcome helped to contribute new categories to Wikipedia and reduced human efforts. Our method is helpful for translating category names in a multilingual scale, which we will continue to work with other languages in the future. To achieve the precision higher, we will apply Google Search data to measure the popularity of translated names, especially for the XY-pattern, and depend on the semantic relatedness [14] to infer category names. We continue to collect more category patterns to widen our translation ability and also apply deep learning networks to improve the performance.

References

1. Bøhn, C., Nørvåg, K.: Extracting named entities and synonyms from wikipedia. In: 2010 24th IEEE International Conference on Advanced Information Networking and Applications. pp. 1300–1307 (2010)
2. Chernov, S., Iofciu, T., Nejdl, W., Zhou, X.: Extracting semantics relationships between wikipedia categories. SemWiki , vol. 206 (2006)
3. Dao, T. N., Simpson, T.: Measuring similarity between sentences. The Code Project (2005)
4. Erxleben, F., Günther, M., Krötzsch, M., Mendez, J., Vrandečić, D.: Introducing wikidata to the linked data web. In: International semantic web conference. pp. 50–65. Springer (2014)
5. Good, B. M., Burgstaller-Muehlbacher, S., Mitraka, E., Putman, T., Su, A. I., Waagmeester, A.: Opportunities and challenges presented by wikidata in the context of biocuration. In: ICBO/BioCreative. Citeseer (2016)
6. Hung, B. T., Le Minh, N., Shimazu, A.: Sentence splitting for vietnamese-english machine translation. 2012 Fourth International Conference on Knowledge and Systems Engineering pp. 156–160 (2012)
7. Koehn, P., Och, F. J., Marcu, D.: Statistical phrase-based translation. Tech. rep., University of Southern California. Information Sciences Institute (2003)
8. Liu, K., Xu, L., Zhao, J.: Opinion target extraction using word-based translation model. In: Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning. pp. 1346–1356 (2012)
9. Luong, M.-T., Manning, C. D.: Achieving open vocabulary neural machine translation with hybrid word-character models. arXiv preprint arXiv:1604.00788 (2016)
10. Nastase, V., Strube, M.: Decoding wikipedia categories for knowledge acquisition. In: American Association for Artificial Intelligence. , vol. 8, pp. 1219–1224 (2008)
11. Pasupat, P., Liang, P.: Zero-shot entity extraction from web pages. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. , vol. 1, pp. 391–401 (2014)
12. Ponzetto, S. P., Strube, M., et al.: Deriving a large scale taxonomy from Wikipedia. In: American Association for Artificial Intelligence. , vol. 7, pp. 1440–1445 (2007)

13. Pu, X., Mascarell, L., Popescu-Belis, A., Fishel, M., Luong, N. Q., Volk, M.: Leveraging compounds to improve noun phrase translation from chinese and german. Proceedings of the ACL-IJCNLP 2015 Student Research Workshop pp. 8–15 (2015)
14. Radhakrishnan, P., Varma, V.: Extracting semantic knowledge from wikipedia category names. In: Proceedings of the 2013 workshop on Automated knowledge base construction. pp. 109–114 (2013)
15. Sarabadani, A., Halfaker, A., Taraborelli, D.: Building automated vandalism detection tools for wikidata. In: Proceedings of the 26th International Conference on World Wide Web Companion. pp. 1647–1654 (2017)
16. Vrandečić, D., Krötzsch, M.: Wikidata: a free collaborative knowledgebase. Communications of the ACM , vol. 57, no. 10, pp. 78–85 (2014)
17. Wang, C., Fan, Y., He, X., Zhou, A.: Learning fine-grained relations from chinese user generated categories. pp. 2577–2587 (2017)
18. Zesch, T., Gurevych, I.: Analysis of the Wikipedia category graph for NLP applications. In: Proceedings of the Second Workshop on TextGraphs: Graph-Based Algorithms for Natural Language Processing. pp. 1–8 (2007)

Electronic edition
Available online: http://www.rcs.cic.ipn.mx

ISSN: 1870-4069
http://rcs.cic.ipn.mx

Centro de Investigación
en Computación