Answer Extraction for Definition Questions using Information Gain and Machine Learning

Carmen Martínez-Gil[1] and A. López-López[2]

Abstract. Extracting nuggets (pieces of an answer) is a very important process in question answering systems, especially in the case of definition questions. Although there are advances in nugget extraction, the problem is finding some general and flexible patterns that allow producing as many useful definition nuggets as possible. Nowadays, patterns are obtained in manual or automatic way and then these patterns are matched against sentences. In contrast to the traditional form of working with patterns, we propose a method using information gain and machine learning instead of matching patterns. We classify the sentences as likely to contain nuggets or not. Also, we analyzed separately in a sentence the nuggets that are *left* and *right* of the target term (the term to define). We performed different experiments with the collections of questions from the TREC 2002, 2003 and 2004 and the F-measures obtained are comparable with the participating systems.

1 Introduction

Question Answering (QA) is a computer-based task that tries to improve the output generated by Information Retrieval (IR) systems. A definition question is a kind of question whose answer [12] is a complementary set of sentence fragments called nuggets.

After identifying the correct target term (the term to define) and context terms, we need to obtain useful and non redundant definition nuggets. Nowadays, patterns are obtained manually as surface patterns [7]. Also, patterns are very rigid,

[1] Carmen Martínez-Gil

Instituto Nacional de Astrofísica Óptica y Electrónica, Facultad de Ciencias de la Computación, Universidad de la Sierra Juárez,, email: carmen@inaoep.mx

[2] A. López-López

Instituto Nacional de Astrofísica Óptica y Electrónica, Luis Enrique Erro #1 Santa María Tonantzintla, 72840 Puebla, México, email: allopez@inaoep.mx

other case can be a soft pattern [4], even also extracted in an automatic way [5]. Then, once we have the patterns we apply a matching process to extract the nuggets. Finally, we need to perform a process to determine if these nuggets are part of the definition; where a common criterion employed is the repetition of the nugget.

According to the state of the art the F-measure in a pilot evaluation [12] for definition questions in 2002 is 0.688 using the nuggets set of author and 0.757 using the nuggets set of other with β=5. For the TREC 2003 [13] F-measure is 0.555 with β=5 and the TREC 2004 [14] F-measure is 0.460 with β=3.

In contrast to the traditional way to extract nuggets, we propose a method that uses two approaches: information gain and machine learning (ML), in particular Support Vector Machine (SVM), Random Forest (RF) and k-nearest-Neighbor (K-NN). We extract the sentence fragments to the *left* and *right* of the target term in an automatic way. These sentence fragments are obtained using a parser (Link Grammar) in the relevant sentences. Then, from parsed sentence we obtained four kinds of sentences fragments, noun phrase containing an appositive phrase, noun phrase containing two noun phrases separated by comma, embedded clauses, and main or subordinate clauses without considering embedded clauses. For the machine learning approach, we labeled with the correct tag, *positive* if the nugget is part of the definition and *negative* otherwise, to prepare the training set of a classifier. So, when we have a sentence fragment and we want to determine if it defines the target term, we apply the classifier.

For this task we work with the questions of the pilot evaluation of definition questions 2002, TREC 2003 and TREC 2004. First, we test each approach, i.e. frequencies, information gain and machine learning algorithms. Then, we combine the sentence fragments obtained with information gain and the sentence fragments labeled classified like *positive* by the ML algorithms.

The paper is organized as follows: next section describes the process to extract sentence fragments; Section 3 describes the approaches used and the method to retrieve only definition sentence fragments; Section 4 reports experimental results; some conclusions and directions for future work are presented in Section 5.

2 Sentence Fragments Extraction

Official definition of F-measure used in the TREC evaluations [12] is:
Let *r* # of vital nuggets returned in a response
a # of non-vital nuggets returned in a response
R total # of vital nuggets in the assessors' list
l # of non-whitespace characters in the entire answer string
Then

$$recall(\Re) = r / R \tag{1}$$

$$allowance(\alpha) = 100 \times (r + a) \tag{2}$$

$$precision(\mathrm{P}) = \begin{cases} 1 & if \quad l < \alpha \\ 1 - \dfrac{l - \alpha}{l} & otherwise \end{cases} \tag{3}$$

Finally, the $F(\beta = 3) = \dfrac{(\beta^2 + 1) \times \mathrm{P} \times \Re}{\beta^2 \times \mathrm{P} + \Re}$ (4)

So, a reason to extract sentence fragments is that we need to retrieve only the most important information from relevant sentences. Other reason to extract short sentence fragments is related to the performance F-measure applied to definition systems in the TREC evaluation; this measure combines the recall and precision of the system. The precision is based on length (in non-white-space characters) used as an approximation to nugget precision. The length-based measure starts from an initial allowance of 100 characters for each (vital or no-vital) nugget matched. Otherwise, the measure value decreases as the length the sentence fragment increases.

We use Lucene [15] system to extract candidate paragraphs from the AQUAINT Corpus of English News Text. From these candidate paragraphs we extract the relevant sentences, i.e. the sentences that contain the target term. Then, to extract sentence fragments we proposed the following process:

1) Parse the sentences. Since we need to obtain information segments (phrases or clauses) from a sentence, the relevant sentences were parsed with Link Grammar [6]. We replace the target by the label **SCHTERM**. For example the sentence for the target term **Carlos the Jackal**:

The man known as **Carlos the Jackal** has ended a hunger strike after 20 days at the request of a radical Palestinian leader, his lawyer said Monday.

The Link Grammar produces:

```
[S [S [NP [NP The man NP] [VP known [PP as [NP
SCHTERM NP] PP] VP] NP] [VP has [VP ended [NP a hunger
strike NP] [PP after [NP 20 days NP] PP] [PP at
[NP [NP the request NP] [PP of [NP a radical Palestinian
leader NP] PP] NP] PP] VP] VP] S] , [NP his
lawyer NP] [VP said [NP Monday NP] . VP] S]
```

2) Resolve co-references. We want to obtain main clauses without embedded clauses or only embedded clauses, so we need to resolve the co-reference, otherwise important information can be lost. To resolve co-reference the relative pronouns WHNP are replaced with the noun phrase preceding it.

3) Obtain sentence fragments. An information nugget or an atomic piece of information can be a phrase or a clause. We analyzed the sentences parsed with Link Grammar and we have identified four kinds of sentence fragments directly related to the target with a high possibility that their information define the target:

a) *Noun phrase (NP) containing an appositive phrase.*
b) *Noun phrase (NP) containing two noun phrases separated by comma [NP, NP].*
c) *Embedded clauses (SBAR).*
d) *Main or subordinate clauses (S) without considering embedded clauses.*

To retrieve the four kinds of sentence fragments we analyze the tree following this procedure:

I. Looking for the nodes which contain the target, in our case the label SCHTERM.
II. Find the initial node of the sentence fragment. The process analyzes the path from the node with the SCHTERM label towards the root node. The process stops when a NP with appositive phrase, NP with [NP, NP], an embedded clause SBAR, or a clause S is reached.
III. Retrieve the sentence fragment without embedded clauses.
IV. Mark as visited the parent node of the second phrase. In case [NP1, NP2] mark as visited the parent node of NP2. For appositive phrase, SBAR or S, the second phrase can be NP, VP or PP.

The steps II – IV are repeated for the same node with a SCHTERM label until a visited node is found in the path to the node towards the root node or the root node is reached. Also the steps II – IV are repeated for each node found in step I.

The next module of our definition question system selects definition sentence fragments. In order to select only definition nuggets from all of sentence fragments, we analyze separately, the information that is to the left of SCHTERM and the information that is to the right of SCHTERM, so we form two data sets.

Now, we present some sentence fragments of two sets obtained using the process for the target term **Carlos the Jackal**:

```
Right sentence fragments
SCHTERM , a Venezuelan serving a life sentence in a French prison
SCHTERM , nickname for Venezuelan born Ilich Ramirez Sanchez
SCHTERM , is serving a life sentence in France for murder
SCHTERM as a comrade in arms in the same unnamed cause
SCHTERM refused food and water for a sixth full day
SCHTERM , the terrorist imprisoned in France

Left sentence fragments
the   friendly   letter   Chavez   wrote   recently   to   the   terrorist
SCHTERM
The defense lawyer for the convicted terrorist known as SCHTERM
he was harassed by convicted international terrorist SCHTERM
an accused terrorist and a former accomplice of SCHTERM
Ilich Ramirez Sanchez , the terrorist known as SCHTERM
Ilich Ramirez Sanchez , the man known as SCHTERM
```

Analyzing separately the sentence fragments before and after the target term is an advantage since in many candidate sentences only one part contains information that defines the target term.

3 Nuggets Selection

In order to obtain only the informative nuggets from the *left* and *right* sentence fragments we use two approaches, one using statistical methods and the other using machine learning algorithms. In the statistical methods we assess the information gain of each fragment and simple frequencies. For the latter we only obtained word frequencies for the sake of comparison. We describe information gain and the machine learning algorithms.

3.1 Information Gain

The information gain [2] for each word or term *l* is obtained using the following definition:

Given a set of sentence fragments *D*, the entropy *H* of *D* is:

$$H(D) \equiv \sum_{i=1}^{c} - p_i \log_2 p_i \quad (5)$$

Where *Pi* is the probability of *i* word and *c* is the size of the vocabulary. Now, for each term *l*. Let D^+ be the subset of sentence fragments of *D* containing *l* and D^- its complement. The information gain of *l*. *IG(l),* is defined by

$$IG(l) = H(D) - \left[\frac{|D^+|}{|D|} H(D^+) + \frac{|D^-|}{|D|} H(D^-) \right] \quad (6)$$

3.2 Machine Learning Algorithms

The other approach to determine if a sentence fragment is part of the definition is using a machine learning algorithm, if it is labeled like positive, then is part of a definition sentence. The ML algorithms that we used are Support Vector Machine, Random Forest, and k-Nearest-Neighbors. We describe briefly each algorithm in the following sections.

Support Vector Machine

The Support Vector Machine (SVM) is a classification technique developed by Vapnik [3], [11]. The method conceptually implements the idea that input vectors are no-linearly mapped to a very high-dimension feature space. In this feature space a linear decision surface is constructed. Special properties of the decision surface ensure high generalization ability of the learning machine. The main idea behind the technique is to separate the classes with a surface that maximizes the margin between them. SVM is based on the Structural Risk Minimization (SRM) principle [11] from computational learning theory. We used a polynomial kernel to perform our experiments.

Random Forest

Random Forest [1] is a classifier that consists of several decision trees. The method uses Breiman's bagging idea and Ho's random subspace method [8] to construct a collection of decision trees with controlled variations. Random forests are a combination of tree predictors such that the tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.

K-Nearest-Neighbor

K-Nearest-Neighbor (K-NN) belongs to the family of instance-based learning algorithms. These methods simply store the training examples and when a new query instance is presented to be classified; its relationship to the previously stored examples is examined in order to assign a target function value. A more detailed description of this algorithm can be found in [9]. In this work, we use distance-weighted K-NN.

3.3 Method to Select Nuggets

To obtain informative nuggets we combine two processes, one using information gain and the other using machine learning algorithms. The process that uses information gain is the following:

I) Obtain the vocabulary of all the sentence fragments (*left* and *right* sets).
II) Obtain the information gain for each word of the vocabulary using the definition of section 3.1.
III) Using the value of the information gain of each word (except stop words), calculate the sum of each sentence fragment.
IV) Rank the sentences fragments according to the value of the sum.
V) Eliminate redundant sentence fragments.

To eliminate redundancy, we compare pairs (*X, Y*) of sentence fragments using the following steps:

a) Obtain the word vector without empty words for each sentence fragment.
b) Find the number of identical words between the two Sentence Fragments *SF*.

c) If $\frac{SF}{|X|} \geq \frac{2}{3}$ or $\frac{SF}{|Y|} \geq \frac{2}{3}$, remove the sentence fragment with lower sum of information gains of the vector. We tested others thresholds but with 2/3 we obtained the better results.

To illustrative the process to eliminate redundancy, we present the following sentence fragments for the target **Carlos the Jackal,** with their corresponding sums:

```
2.290 nickname for Venezuelan born Ilich Ramirez Sanchez
2.221 Ilich Ramirez Sanchez , the Venezuelan born former guer-
rilla
2.157 Ilich Ramirez Sanchez , the terrorist
1.930 Ilich Ramirez Sanchez , the man
1.528 Illich Ramirez Sanchez
```

If we compare the first and the second sentences, the result of the step a) is:

```
[nickname, Venezuelan, born, Ilich, Ramirez, Sanchez]
[Ilich, Ramirez, Sanchez, Venezuelan, born, former, guerrilla]
```

In the step b) we obtained that SW=5.

Finally, in the step c) we remove the second sentence fragment since it has a lower sum of information gains. Applying the procedure with the other sentence fragments, the result is that we keep only the first:

2.290 nickname for Venezuelan born Ilich Ramirez Sanchez

For the machine learning algorithms we apply the following process. From the AQUAINT Corpus and following the process described in the section 2, we obtained the sentence fragments to form two training sets for the three learning algorithms. The *left* set contains 2982 examples and the *right* set contains 3681 examples. The sets were formed with a ratio of 1:3 between positive and negative examples in order to have balanced sets. One sentence fragment was labeled as *positive* if it contains information of a vital or no vital nugget and *negative* otherwise. The sentence fragments were tagged with POS [10]. Then, we maintain the two words closer to the target term and the following five tags, so a window of seven words and tags is obtained. We tested others combinations likes all labels POS or maintain the word closer to the target but the best result was obtained maintain the two words closer.

An illustrative example to obtain the training set for the target **Christopher Reeve**, using only three sentences fragments, is the following:

Right set of sentence fragments

```
SCHTERM is paralyzed from a spinal cord injury in a rid-
ing accident
SCHTERM, the actor confined to a wheelchair from a
horseback riding accident
SCHTERM told a 6 year old girl paralyzed in an amusement
park accident
```

Sentence fragments tagged with POS

```
SCHTERM/NNP is/VBZ paralyzed/VBN from/IN a/DT spinal/JJ
cord/NN injury/NN in/IN a/DT riding/VBG accident/NN
```

```
SCHTERM/NNP ,/, the/DT actor/NN confined/VBD to/TO a/DT
wheelchair/NN from/IN a/DT horseback/NN riding/VBG acci-
dent/NN
SCHTERM/NNP told/VBD a/DT 6/CD year/NN old/JJ girl/NN
paralyzed/VBN in/IN an/DT amusement/NN park/NN acci-
dent/NN
```

Final coding for training set

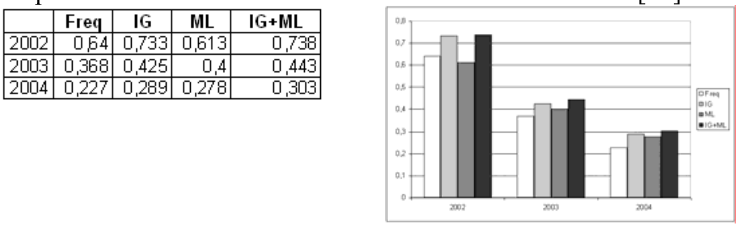
```
is, paralyzed, IN, DT, JJ, NN, NN, POSITIVE
COMMA, the, NN, VDB, TO, DT, NN, POSITIVE
told, a, CD, NN, JJ, NN, VBN, POSITIVE
```

4 Experiments Results

We performed experiments with three sets of definition question, the questions from the pilot evaluation 2002, TREC 2003 and TREC 2004. (We did not compare our results with the collections of the TREC 2005 and 2006 since in these years the list of nuggets was not readily available). First we test each approach, i.e. frequencies, information gain and the machine learning algorithms. For the latter approach we used the training set described in the section 3.3 but excluding from the training set the collection on evaluation. Then, we combine the sentence fragments obtained with information gain and the sentence fragments classified like *positive* by the machine learning algorithms.

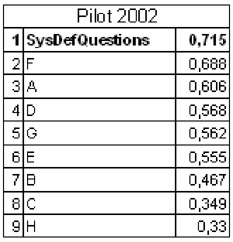
Values of the F-measure are shown in the figure 1 and Freq is the baseline. In every set of questions, information gain obtained higher F-measure than simple frequencies and machine learning algorithms. But the best value of the F-measure is obtained when we combined information gain with the machine learning algorithms, since the two approaches are complementary, the first approach obtained the most frequent sentence fragments and the second approach retrieves the information that has implicitly or explicitly a definition pattern.

It is important to note that with the collection 2002 there are two set of nuggets AUTHOR and OTHER. We compare the output of our system (labeled SysDefQuestions) with the set's AUTHOR nuggets. Figure 2 shows the comparison of F-measure values obtained in the pilot evaluation version of definition questions using the AUTHOR set of nuggets [12]. The figure 4 shows the comparison of F-measure values obtained in the TREC 2003 [13]. Finally, in the figure 5 we present the comparison of F-measure values obtained in the TREC 2004[14].



	Freq	IG	ML	IG+ML
2002	0,64	0,733	0,613	0,738
2003	0,368	0,425	0,4	0,443
2004	0,227	0,289	0,278	0,303

Fig. 1. Comparison of the F-measures obtained with Frequencies Freq, information gain IG, machine learning algorithms ML, and the combination of IG with ML.



Pilot 2002		
1	**SysDefQuestions**	**0,715**
2	F	0,688
3	A	0,606
4	D	0,568
5	G	0,562
6	E	0,555
7	B	0,467
8	C	0,349
9	H	0,33

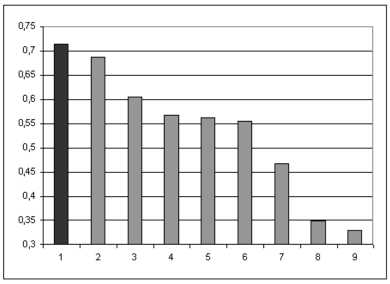
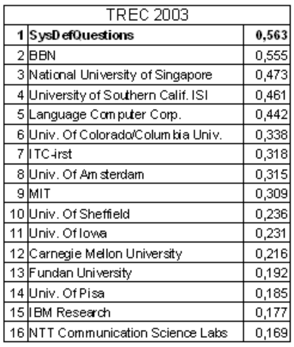


Fig. 2. Comparison of F-measure values of pilot evaluation of definition questions using the AUTHOR list of nuggets



TREC 2003		
1	**SysDefQuestions**	**0,563**
2	BBN	0,555
3	National University of Singapore	0,473
4	University of Southern Calif. ISI	0,461
5	Language Computer Corp.	0,442
6	Univ. Of Colorado/Columbia Univ.	0,338
7	ITC-irst	0,318
8	Univ. Of Amsterdam	0,315
9	MIT	0,309
10	Univ. Of Sheffield	0,236
11	Univ. Of Iowa	0,231
12	Carnegie Mellon University	0,216
13	Fundan University	0,192
14	Univ. Of Pisa	0,185
15	IBM Research	0,177
16	NTT Communication Science Labs	0,169

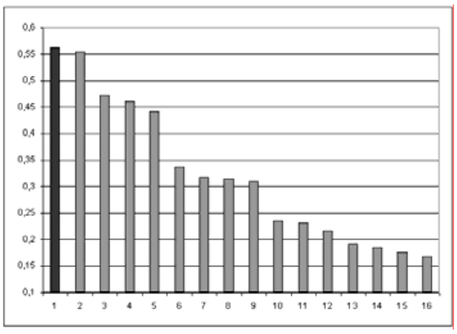
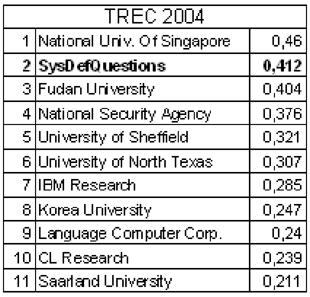


Fig. 3. Comparison of F-measure values of TREC 2003.



TREC 2004		
1	National Univ. Of Singapore	0,46
2	**SysDefQuestions**	**0,412**
3	Fudan University	0,404
4	National Security Agency	0,376
5	University of Sheffield	0,321
6	University of North Texas	0,307
7	IBM Research	0,285
8	Korea University	0,247
9	Language Computer Corp.	0,24
10	CL Research	0,239
11	Saarland University	0,211

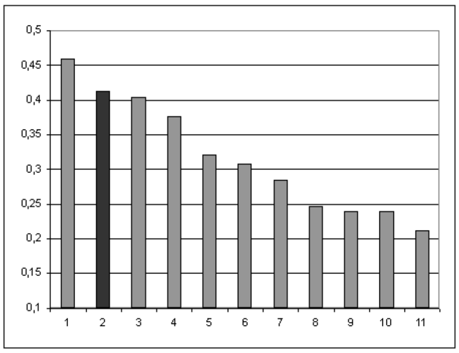


Fig. 4. Comparison of F-measure values of TREC 2004.

From two sets of definition questions, we can observe that our system SysDefQuestions retrieves most of the definition sentence fragments. For the set of definition questions of TREC 2004 the F-measure of our system is competitive when compared to the participating systems.

5 Conclusions and Future Works

We have presented a method to extract definition sentence fragments called nuggets in an automatic and flexible way and the results obtained are comparable with the participating systems in the TREC. The sentence fragments obtained with the process presented are acceptable since these contain only the information directly related to the target. Other advantage is that these sentence fragments present a short length, and this improves the precision of our definition question system.

We are planning to categorize the targets in three classes: ORGANIZATIONS, PERSON and ENTITIES and then train three different classifiers.

Acknowledgments: The first author was supported by scholarship 157233 granted by CONACyt, while second author was partially supported by SNI, Mexico.

References

1. Breiman, L.: Random Forest. Machine Learning 45 (1), (2001) 5-32.
2. Carmel, D., Farchi, E., Petruschka, Y., and Soffer, A.: Automatic Query refinement using lexical affinities with maximal information gain. *SIGIR* (2002): 283-290.
3. Cortes, C. and Vapnik, V.: Support Vector Networks. Machine Learning. (1995) 20:1-25.
4. Cui, H., Kan, M. Chua, T. and Xiao, J.: A Comparative Study on Sentence Retrieval for Definitional Questions Answering. SIGIR Workshop on Information Retrieval for Question Answering (IR4QA), (2004) 90-99.
5. Denicia-Carral, C., Montes-y-Gómez, M., and Villaseñor-Pineda, L.: A Text Mining Approach for Definition Question Answering. 5th International Conference on Natural Language Processing, Fin Tal. Lecture Notes in Artificial Intelligence, Springer (2006).
6. Grinberg, D., Lafferty, J., and Sleator, D.: A robust parsing algorithm for link grammars. Carnegie Mellon University Computer Science technical report CMU-CS-95-125, and *Proceedings of the Fourth International Workshop on Parsing Technologies*, Prague, September, (1995).
7. Hildebranddt, W., Katz, B. and Lin, J.: Answering Definition Question Using Multiple Knowledge Sources. In Proceeding of HLT/NAACL, Boston (2004) 49-56.
8. Ho, T.: The Random Subspace Method for Constructing Decision Forests. IEEE Trans. on Pattern Analysis and Machine Intelligence 20 (8), (1998) 832-844.
9. Mitchell, T.: *Machine Learning*. McGraw-Hill. (1997).
10. Toutanova, K., Klein, D., Manning, C., and Singer, Y.: Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. *In Proceedings of HLT-NAACL* (2003): 252-259.
11. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York. (1995).
12. Voohees, E.: Evaluating Answering to Definition Questions. NIST (2003) 1-3.
13. Voorhees, E.: Overview of the TREC 2003 Question Answering Track. *NIST* (2003): 54-68.
14. Voorhees, E.: Overview of the TREC 2004 Question Answering Track. *NIST* (2004): 12-20.

http://lucene.apache.org/java/docs/