Automatic Multiple Choice Question Generation System for Semantic Attributes Using String Similarity Measures

Ibrahim Eldesoky Fattoh
Teacher Assistant at faculty of Information Technology, MUST University, Cairo, Egypt.

Abstract

This research introduces an automatic multiple choice question generation system to evaluate the understanding of the semantic role labels and named entities in a text. The system provided selects the informative sentence and the keyword to be asked based on the semantic labels and named entities that exist in the sentence, the distractors are chosen based on a similarity measure between sentences in the data set. The system is tested using a set of sentences extracted from the TREC 2007 dataset for question answering. From the experimental results, it can be induced that the semantic role labeling and named entity recognition approaches could be used as a good keyword selection mechanism. The second conclusion is that the string similarity measures proved to be a very good approach that can used in generating the distractors for an automatic multiple choice question. Also, combining the similarity measures of different algorithms would lead to generate a good distractors.

1. Introduction

Developing Automatic Question Generation (AQG) systems became one of the important research issues because it requires insights from a variety of disciplines, including, Artificial Intelligence (AI), Natural Language Understanding (NLU), and Natural Language Generation (NLG). There are two types of question formats; multiple choice questions which asks about a word in a given sentence, the word may be an adjective, adverb, vocabulary, etc., the second format is the entity questions systems or Text to Text QG that asks about a word or phrase corresponding to a particular entity in a given sentence. In this research the first type of question formats is covered. The traditional multiple-choice question is made up of three components, where the sentence with a gap is defined as the question sentence, the correct choice (removed word) as the key, and the other alternative choices as the distractors [1].

- _______________ is the current president of Egypt
  (a) H.Mubarak (b) A.ELsisi (c) M.Morsi (d)A.Mansour

The above sentence is an example of multiple choice question, the underline gap represents the word or phrase that is the correct answer, the four choices represent the true answer and three distractors . This research introduces a model for a multiple choice question generator that asks about labels extracted from the given sentence using Semantic Role Labeler (SRL) and entities extracted using Named Entity Recognizer (NER). The distractors generated for the sentence are chosen based on the string similarity between the question sentence and all other sentences in the data set. The rest of the paper is organized as follows: section 2 discusses the related work of Automatic Multiple Choice Questions (AMCQ), section 3 introduces the SRL and NER in brief, section 4 provides the different string text similarity approaches, section 5 introduces the proposed model, and section 6 shows the experimental results and evaluation, and finally section 7 introduces a conclusion and future work with some remarks.

2. Related work

In this section, a review of the previous Automatic Multiple Question Generation systems for the first question type formats mentioned in section 1 is introduced.

Authors in [2] proposed an approach for AQG for vocabulary assessment; they generated 6 types of questions: definition, synonym, antonym, hypernym, hyponym, and cloze questions. They retrieve the data from WordNet after choosing the correct sense for it. Concerning the distractor choice, the question generation system chooses distractors of the same part of speech and similar frequency to the correct answer. Four of the six computer-generated question types were assessed: the definition, synonym, antonym, and cloze questions. The percentage of questions generated for the four types were above 60% for 156 word list.

The authors of [3] introduced a prototype for an automatic quiz generation system for English text to test learner comprehension of text content and English skills. They used the semantic network to represent the relationship between a vocabulary and its context. They proposed two generators for two types of questions. The first generator is for sense comprehension of adjectives; the generator will extract adjectives from the SemNet of a given text as questionnaire vocabularies and form multiple-choice cloze questions. The right answer is substituted by the synonym or a similar adjective of the applied sense of the questionnaire adjective from WordNet. The second generator is for anaphor comprehension, a learner must integrate these subnets by connecting each anaphor with its antecedents. The generator identifies the antecedent of an anaphor and form a multiple-choice cloze question by scooping the anaphor out of its sentence. The options comprise its antecedent and the distractors.

The same authors of [3] proposed another research [4] for multiple choice questions for understanding the evaluation of adjectives in a text. Also based on the sense association among adjectives, an adjective being examined can be usually substituted by some other adjectives. The system was able to generate three types of questions: questions for collocations, questions for antonyms, and questions for synonyms. For a given sentence, the system extracts an adjective-noun pairs that exist, then for each adjective-noun pair, if it is a collocation, generate a question for it. If the original sentence has words which have negative meanings, generate a question for antonyms. Also generate questions for synonyms or similar words. The candidates of a substitute are gathered from WordNet and filtered by web corpus searching. For evaluating the generated questions, they choose Far East senior high school English textbook, Book One, which contains 12 articles, as the experimental material. Experimental results have shown that the proposed answer determination approaches and question filtering strategies are effective in precision.

Another automatic question generation system that can generate gap-fill questions provided by [5]. Syntactic and lexical features are used in the process of choosing the informative sentence, determining the key, and finding the distractors. The authors introduced some features as a basis for sentence selection like its position, common tokens, contains an abbreviation and others. In the key selection, part of speech tagging (POS) used to generate a list of keys, then selecting the best key from this list depend on three parameters which are; number of occurrences of the key in the document, does it is a word in the title, and height of the key in the syntactic tree. The distractor selection depends on some features like Dice coefficient score between gap fill sentence and the sentence containing the distractor and others. The system was tested using two chapters of the biology book and has been evaluated manually by two biology students. The sentence selection module takes 0.7 inter evaluator agreement, the key selection takes 0.75 inter evaluator agreement, and 0.60 are useful gap fill question which has at least one good distractor.

From this literature review, it can be noted that building an automatic multiple choice question generation system concerns with three steps, the first is choosing the informative sentence, the second is choosing the key word or phrase to be the right answer in the multiple choices, and the last is finding the distractors for that key word. In this research, the informative sentence selection depends on if the sentence contains any named entities or semantic labels. Also the keys that are chosen will base on the output of semantic role labeling and named entity recognizer. And the distractors selection will be based on the string based similarity measures as will be explained in section 4.

3. Semantic Role Labeling (SRL) and Named Entity Recognition (NER)

Semantic role labeling describe WHO did WHAT to WHOM, WHEN, WHERE, WHY, HOW etc. for a given situation, and contribute to the construction of meaning [6], for this reason the natural language processing community has recently experienced a growth of interest in SRL. SRL has been used in many different applications like automatic text summarization [6] and automatic question answering [7]. Given a sentence, a semantic role labeler tries to identify the predicates (relations and actions) and the semantic entities associated with each of those predicates. The set of semantic roles used in PropBank [8] includes both predicate-specific roles whose precise meaning are determined by their predicate, and general-purpose adjunct-like modifier roles whose meaning is consistent across all predicates. The predicate specific roles are Arg0,Arg1, ..., Arg5 and ArgA. A complete list of the modifier roles as proposed in the PropBank are shown in table 1. Giving a sentence like

Anders Celsius born in Uppsala in Sweden *(1)*

The SRL parse would be as seen in (2).

***[Andres Celsius /A0] [**born /v:] [in Uppsala /AM-Loc] [in Sweden/ AM-Loc]* *(2)*

The relation identified in (2) is the verb (born), the predicate specific roles are (Andres celsius) identified as A0 (Arg 0), is the subject of the verb, and (in Uppsala) identified as AM Location, Also, (in Sweden) identified semantically as AM Location which is a general purpose adjunct.

Table 1: ProbBank Arguments Roles

Role	Meaning
ArgM-LOC	Location
ArgM-EXT	Extent
ArgM-DIS	Discourse connectives
ArgM-ADV	Adverbial
ArgM-NEG	Negation marker
ArgM-MOD	Modal verb
ArgM-CAU	Cause
ArgM-TMP	Temporal
ArgM-PNC	Purpose
ArgM-MNR	Manner
ArgM-DIR	Direction
ArgM-PRD	Secondary prediction

Another set of semantic attributes like persons, organizations, locations, erc.,can be recognized using named entity recognition systems. Named entity recognition is an essential task in many natural language processing applications nowadays, and is given much attention in the research community and considerable progress has been achieved in many domains, such as news wire and biomedical [9]. If we have the sentence in (1), *the output of NER would be like (3).*

*[**Person** Andres Celsuis] born [**Loc** Uppsala] in [**Loc** Sweden].* *(3)*

Entity (Andres Celsuis) is identified as person , both entities (Uppsala) and (Sweden) are identified as location. All these entities could be used as a target by replacing them with gaps, one at a time. The attributes extracted from both NER and SRL act as the keywords which we search for in the sentence to be asked for are shown in table 2.

Table 2: Keyword types (labels and entities) selected from the question sentence

Keyword Types	Source
<AM-CAUS>	SRL
<Person>	NER
<AM-LOC>	SRL
<Location>	NER
<AM-TMP>	SRL
<Date>	NER
<Time>	NER

4. Text Similarity Approaches

Text similarity measures play an important role in NLP applications such as text classification, information retrieval, document clustering, short answer scoring, machine translation, text summarization and others. Finding the similarity between words is a fundamental step in finding the similarity between sentences and documents [10]. Words can be lexically similar and semantically similar. The words are lexically similar in case of they share the similar sequence of characters and they are semantically similar in different cases like if they have the same thing, are opposite of each other, used in the same context and one is a type of another. In this research, a set of the string-based similarity algorithms are applied to measure the similarity between the question sentence and the remaining sentences exist in the knowledge base as a new methodology proposed to choose the distractors for the keyword asked in a multiple choice question. The string metric is a metric that measures the similarity or distance between two strings. The string similarity algorithms are divided into two categories, the first one is the character based similarity algorithms, and the second is the term based similarity algorithms. In this research, three algorithms of the character based type, and five algorithms of the term based types were applied to measure the similarity between two sentences. The character based algorithms used are Smith-Waterman [11], Damerau-Levenshtein [12, 13], and Jaro [14, 15]. The five term-based algorithms applied are N-gram, Cosine similarity, Dice’s coefficient [16], Jaccard similarity [17], and Block distance [18]. These algorithms are explained and implemented in SimMetrics package [19]. Figure 1 illustrates the string based algorithms applied in this research. A survey about these algorithms and text similarity approaches exists in [10].

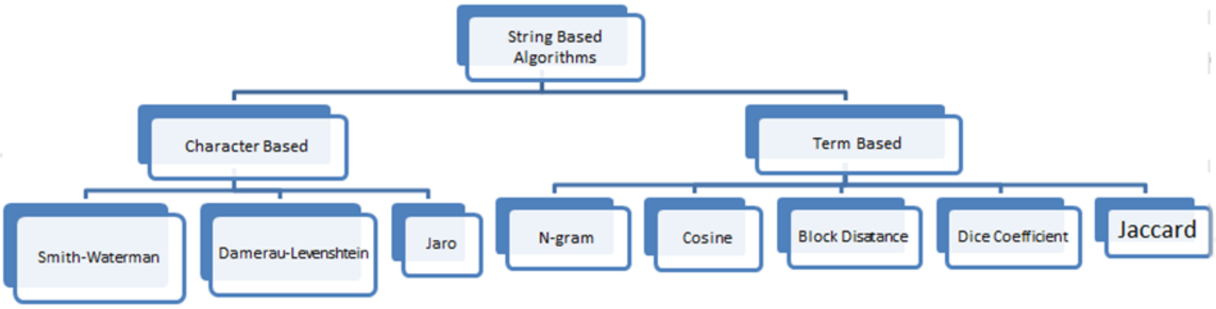

Figure 1: applied algorithms in this research

5. Proposed model

The automatic multiple choice questions system proposed in this research asks for the semantic roles, and the named entities exist in a sentence like attributes specified in table 2. At the beginning, a knowledge base is prepared by extracting the sentences from the used dataset, then parsing them semantically using a semantic role labeling tool and named entity recognizer for discovering the attributes that exist in the sentence. The SENNA tool is used for both purposes [20]. The sentence that has any semantic attribute is recorded in the knowledge base and its attribute is linked with it. To generate a question, the question sentence is chosen from the knowledge base and the keyword asked for is considered the labeled word or entity word identified by SENNA tool and is substituted with a gap. The distractors for the key word asked are considered from the other keyword for the remaining sentences in the knowledge base. To find a distractor a string similarity measure between the question sentence and all other sentences exist within the knowledge base is applied. Then, 3 keywords are retrieved, these keywords belong to the sentences that got the highest similarity values. The retrieved three keywords are considered to be the distractors for the question sentence. Both Algorithm 1 and figure 2 show the basic steps followed in the proposed model.

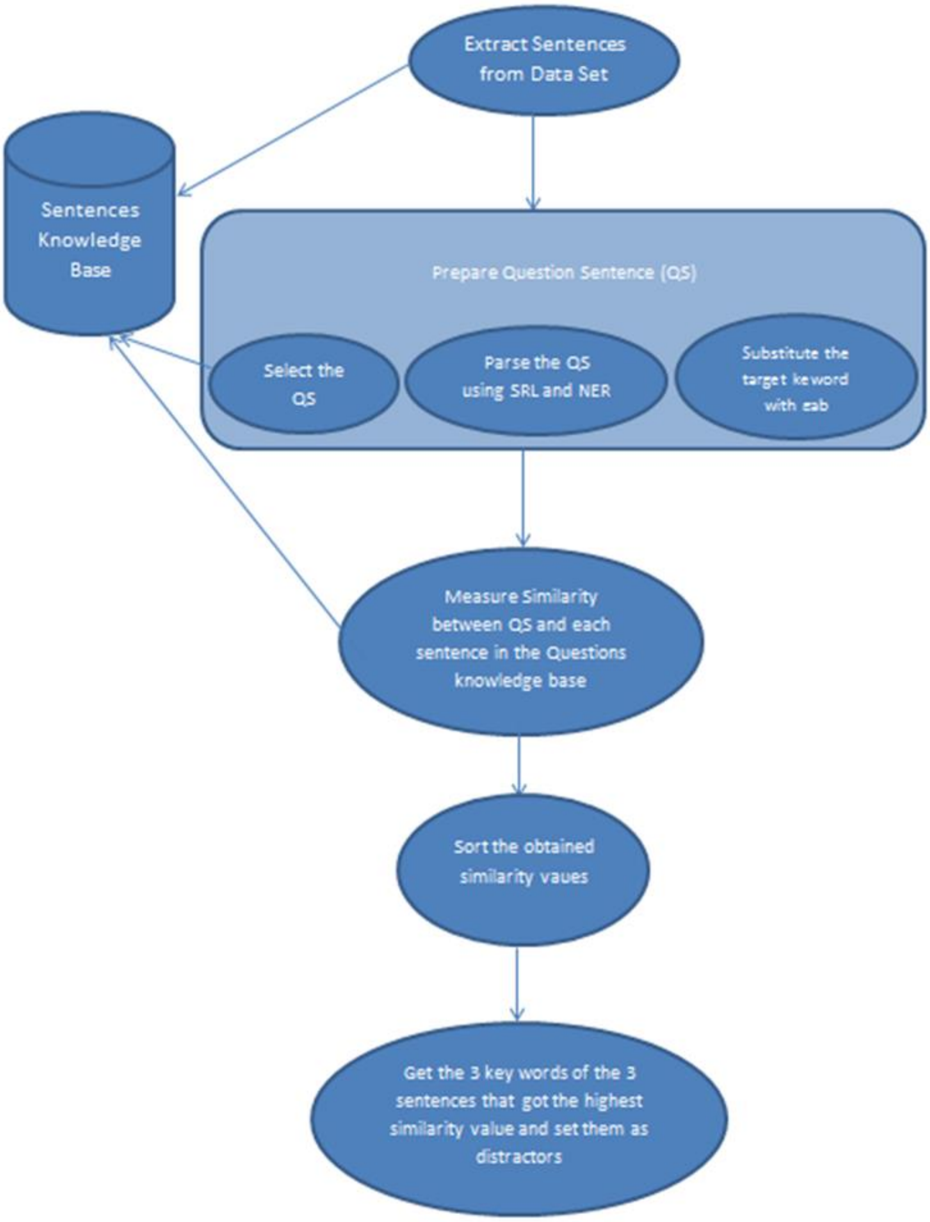
Extract Sentences from Data Set
Sentences Knowledge Base
Prepare Question Sentence (QS)
Select the QS
Parse the QS using SRL and NER
Substitute the target keword with æab
Measure Similarity between QS and each sentence in the Questions knowledge base
Sort the obtained similarity vaues
Get the 3 key words of the 3 sentences that got the highest similarity value and set them as distractors

Figure 2: flow diagram of automatic multiple choice question generation system

```
Algorithm
Build the Knowledge Base by extracting the sentences which have the semantic attributes from the dataset
Select a question sentence and identify the semantic type of the keyword by parsing it semantically
Foreach question sentence
        Measure the similarity between the question sentence and all sentences  in the knowledge base
Sort the obtained similarity values.
Return the three sentences that have the highest similarity values
Return three keywords of the three sentences as  distractors and identify their types.
```

6. Experimental Results and Discussion:

In this section, the applied experimental results will be explained. The dataset used is the TREC 2007 dataset for question answering [21]. A set of files of different domain subjects is parsed and 109 sentences are extracted to be used in testing the proposed model. The semantic attributes for these sentences are similar to types in table 2. The 109 sentences that are chosen are the sentences yielded a good result from the SENNA tool in retrieving their semantic attributes. Some sentences are rejected because of their output from the SENNA tool. The evaluation of both sentence selection and keyword identification depend on the output of the tool used to identify semantic attributes of a sentence. In this research out of nearly 145 parsed sentences, there were 109 considered good according to the keywords that are extracted from them. The distractor evaluation is the important part we tried to contribute in this research, so eight string similarity algorithms are applied trying to generate a good distractors. In this research we tried to evaluate the question difficulty according to the distractors generated. The question difficulties levels considered in this research are very difficult, difficult, intermediate, and easy. These levels are proposed according to the type of the generated distractor word. Each question has a true answer which is the keyword exists in the question sentence and three distractors which are generated from the remaining sentences in the knowledge base. To evaluate the usage of the algorithms in generating the distractors , we suggested four classes for the question difficulty level, the question will be very difficult if the all the generated distractors have the same type of the keyword, the question will be difficult if two of the generated distractors have the same type of the keyword, the question will be intermediate if only one of the generated distractors has the same type of the keyword, and the question will be considered as an easy question if all generated distractors are of different types other than the key word's type. For more illustration, consider the following question sentences in table 3.

Table 3: example of question with different difficulty levels

Difficulty level	**Question Sentence**	**Key word**	**Choices**
Very Difficult	_____________was the sixteenth President of the United States	Abraham Lincoln	(A) Abraham Lincoln (B) Barack Obama (C) Calvin Coolidge (D) Anders Celsius
Difficult	__________ is the sixth largest country in Europe in terms of area	Finland	(A) Abraham Lincoln (B) Finland (C) Russia (D) Switzerland
Intermediate	In _____________ Sadat made a historic visit to Israel, which led to the 1979 peace treaty in exchange for the complete Israeli withdrawal from Sinai	1977	(A) Abraham Lincoln (B) 1973 (C) 1977 (D) Finland
Easy	___________ is the capital of the Republic of Austria and one of the nine states of Austria.	Vienna	(A) Abraham Lincoln (B) June 18 1953 (C) Vienna (D) 1977

According to table 3, the evaluation of the eight algorithms of string similarity is performed and their results are shown in table 4. The first column of the table shows the number of questions yielded in each class. The 45 appears in the first row for the N-gram algorithm means that the system yielded 45 questions having three distractors of the same type of the keyword asked.

Table 4: number of sentences obtained in each class of the 8 algorithms

	N-gram	Smith	Levensh-tein	Jaro	Cosine	Dice coefficien t	Block Distance	Jaccard
No of very difficult questions.	45	42	35	21	41	40	42	42
No of difficult questions.	36	27	36	33	31	30	29	30
No of intermediate questions..	21	24	26	34	19	21	20	19
No of easy questions.	7	16	12	21	18	18	18	18

From table 4, it is clear that N-gram algorithm achieves the highest level of difficulty, it yielded 81 questions in the top difficult levels (very difficult and difficult), and only 28 questions for the intermediate and easy levels. Also the Jaro algorithm achieved the highest level of simplicity in the 8 algorithms, it yielded 55 questions in the

intermediate and easy levels, and 54 in the difficult levels. Another measurement introduced to measure the difficulty level of the generated questions for each algorithm by the following equation
Difficulty level of questions = ((3 * No.of three + 2 * No.of two + 1 * No.of one) / 109) /3
Where No.of three is the number of questions that has 3 distractors which type is the same as the key word type. And the same is for No.of two and No.of one. The overall value is divided by 3 at the end of the equation for normalizing the obtained values to get a percentage value. The value of the difficulty level of questions increases as the amount of difficult questions increase. Table 5 shows the value of the difficulty level of questions generated for each algorithm

Table 5: difficulty level of the generated questions for the 8 algorithms

	N-gram	Smith	Levensh-tein	Jaro	Cosine	Dice coefficient	Block Distance	Jaccard
Difficulty level of questions	69.7%	62.4%	62.1%	48.6%	62.4%	61.5%	62.4%	62.7%

The output resulted in table 5 shows that N-gram algorithm got the highest value and the Jaro algorithm got the lowest value which proofs our conclusion about both algorithms before. By considering a useful multiple choice questions are those which have at least one good distractor, and considering a good distractor is the one which has the same type as the keyword type. Table 6 shows the percentage of good questions that generated from each algorithm according to the questions that have at least one good distractor.

Table 6: percentage of good questions for the 8 algorithms

	N-gram	Smith	Levensh-tein	Jaro	Cosine	Dice coefficient	Block Distance	Jaccard
Percentage of good questions	93.6%	85.3%	89%	80.7%	83.5%	83.5%	83.5%	83.5%

It can be noticed from table 6 that the N-gram algorithm got the highest percentage of good question because the least number of the easy questions it has. Also, the percentage value of all term based algorithms except the N-gram is equal to 83.5%, and the cause of that is all of these algorithms resulted the same number of easy questions as shown in table 4. Another evaluation is introduced by combining the best results obtained from the character based algorithms (Smith Waterman results) with the best results obtained from the term based algorithms (N-gram results) to enhance the results obtained. Also, combining the results obtained from both (N-gram algorithm and Jaccard algorithm), the cause of combining the results of these two algorithms is that they got the highest level of questions value from all 8 algorithms as shown in table 5. Table 7 shows the results yielded by combining the results of two different algorithms.

Table 7: results of combining results of 2 different algorithms

	N-gram+Smith	N-gram+Jaccard
No of Very difficult questions.	42	46
No of difficult questions.	30	32
No of intermediate questions..	26	23
No of easy questions.	11	8
Difficulty level of questions	64.8%	68.8%
Percentage of good questions	89.9%	92.7%

From table 7, it is clear that the values obtained from both (N-gram+Smith) increase the values obtained from the Smith's results only in Difficulty level of questions and Percentage of good questions. Combining the N-gram's results with Jaccard's results yielded an increase of the both values compared to Jaccard's results. Also, we can notice that N-gram results still gives the best after combination.

7. Conclusion and future work

This research introduced an automatic generation of multiple choice questions based on the semantic attributes in the question sentence. The semantic attributes are extracted using both semantic role labeling tool and named entity recognition tool. The distractor generation process introduced based on the string similarity measures between the question sentence and all other sentences existed in a knowledge base of all sentences in the system. Eight algorithms of string based similarity are applied for all sentences and the results obtained are analyzed and introduced with a classification introduced to identify the question difficulty level. All algorithms introduced promising results in the process of generating distractors specially the N-gram algorithm which introduced the highest level of difficulty questions. Also, combining the results of more than one algorithm with each other is tried and the output of this process enhances the difficulty level of some algorithms. In the future we could try semantic similarity measures like corpus-based similarity and knowledge base similarity algorithms. Also, a prior classification of the sentences in the knowledge base according the key word types could be introduced to increase the level of difficulty of the generated questions.

References

[1] Chen, C. Y., Liou, H. C., & Chang, J. S. (2006, July). Fast: an automatic generation system for grammar tests. In *Proceedings of the COLING/ACL on Interactive presentation sessions* (pp. 1-4). Association for Computational Linguistics.

[2] Brown, J., Firshkoff, G. And Eskenazi, M. (2005) Automatic Question Generation For Vocabulary Assessment. Proceedings Of Hlt/Emnlp, 819–826. Vancuver, Canada.

[3] Sung, L., Lin, Y., And Chern, M.(2007). An Automatic Quiz Generation System For English Text. Seventh Ieee International Conference On Advanced Learning Technologies.

[4] Lin, Y., Sung, L., And Chern, M (2007). An Automatic Multiple-Choice Question Generation Scheme For English Adjective Understanding. Workshop On Modeling, Management And Generation Of Problems/Questions In Elearning, The 15th International Conference On Computers In Education (Icce 2007), Pages 137-142, Hiroshima, Japan.

[5] Agarwal , M. And Mannem ,P. (2011). Automatic Gap-Fill Question Generation From Text Books. In Proceedings Of The 6th Workshop On Innovative Use Of Nlp For Building Educational Applications. Portland, Or, Usa. Pages 56-64.

[6] Trandabăţ, D. Using semantic roles to improve summaries.(2007). In *The 13th European Workshop on Natural Language Generation* (p. 164).

[7] Pizzato L., and Molla D. (2008). Indexing on Semantic Roles for Question Answering. Coling 2008: Proceedings of the 2nd workshop on Information Retrieval for Question Answering (IR4QA). pages 74–81. Manchester, UK.

[8] Palmer M., Gildea D., and Kingsbury P. (2005). The proposition bank: An annotated corpus of semantic roles. Computational Linguistics, 31(1):71–106.

[9]Tkachenko M., and Simanovisky A. (2012). Named Entity Recognition: Exploring Features. Proceedings of KONVENS 2012 (Main track: oral presentations), Vienna.

[10] Gomaa W. H. And Fahmy A. A. (2013). A survey of text similarity approaches. *International Journal of Computer Applications*, *68*(13), 13-18.

[11] Smith, F. T. & Waterman, S. M. (1981). Identification of Common Molecular Subsequences, Journal of Molecular Biology 147: 195–197.

[12] Hall, P. A. V. & Dowling, G. R. (1980) Approximate string matching, Comput. Surveys, 12:381-402.

[13] Peterson, J. L. (1980). Computer programs for detecting and correcting spelling errors, Comm. Assoc. Comput. Mach., 23:676-687.

[14] Jaro, M. A. (1989). Advances in record linkage methodology as applied to the 1985 census of Tampa Florida, Journal of the American Statistical Society, vol. 84, 406, pp 414-420.

[15] Jaro, M. A. (1995). Probabilistic linkage of large public health data file, Statistics in Medicine 14 (5-7), 491-8.

[16] Dice, L. (1945). Measures of the amount of ecologic association between species. Ecology, 26(3).

[17] Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. Bulletin dela Soci ét éVaudoise des Sciences Naturelles 37, 547-579.

[18] Eugene F. K. (1987). Taxicab Geometry , Dover. ISBN 0-486-25202-7.

[19] Chapman, S. (2009). Simmetrics: a java & c#. net library of similarity metrics.

[20] Collobert R., Weston J., Bottou L E., Karlen M, Kavukcuoglu K, and Kuksa P. (2011). Natural language processing (almost) from scratch. The Journal of Machine Learning Research, 12:2493–2537, 2011.

[21] http://trec.nist.gov/tracks.html