UPB Scientific Bulletin, Series C, 76(1), 31–44.

AUTOMATED ASSESSMENT OF SELF-EXPLANATIONS

Bogdan OPRESCU[1], Mihai DASCĂLU[2], Ștefan TRĂUȘAN-MATU[3], Philippe DESSUS[4], Maryse BIANCO[5]

Auto-explicațiile sunt verbalizări făcute de un cititor în timpul lecturii unui text pentru a-l înțelege mai bine. Sistemul implementat este proiectat să analizeze în mod automat aceste explicații, permițându-i astfel unui profesor să evalueze mai în detaliu nivelul de înțelegere al materialelor citite de elevi. Metoda propusă se bazează pe tehnici specifice de prelucrare a limbajului natural adaptate pentru limba franceză și se adresează utilizării în clasele din școala primară. În plus, în cadrul procesului de analiză a fost integrată o euristică proprie, la nivel de cuvinte, pentru a putea evalua similaritatea dintre textele inițiale și verbalizările elevilor.

Self-explanations are verbalizations that readers give to themselves while reading a text, in order to better understand it. Our implemented system is designed for automatically analyzing self-explanations in order to allow teachers to better grasp the comprehension of pupils of the previously read materials. Our method uses specific natural language processing techniques for French language and it is conceived for use with primary school pupils. Furthermore, we have integrated a word-based heuristic in order to measure similarity between the initial texts and the pupil's verbalizations.

Keywords: Self-explanations, Verbalizations, Self-Explanation Reading Training (SERT), Latent Semantic Analysis (LSA), Automated assessment

1. Introduction

Psychological and pedagogical research has revealed that people tend to understand better a text if they try to explain themselves what they have read [1], [2]. Starting from these observations, techniques, such as SERT (Self-Explanation

[1] Master's Degree Student, Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: bogdan.oprescu@cti.pub.ro
[2] Teaching Assistant, Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: mihai.dascalu@cs.pub.ro
[3] Professor, Dept. of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: stefan.trausan@cs.pub.ro
[4] Professor, Lab. Sciences de l'Education, Université Pierre-Mendès-France, France, e-mail: philippe.dessus@upmf-grenoble.fr
[5] Associate Professor, Lab. Sciences de l'Education, Université Pierre-Mendès-France, France, e-mail: maryse.bianco@upmf-grenoble.fr

Reading Training) [3], were developed to help pupils understand texts and to make the learning process more efficient and focused on comprehension.

The macro-script used within our educational experiments consisted of the following: at predefined moments, pupils were asked, during their reading, to stop and explain what they had understood up to that moment. Their explanations were recorded and later on transcribed, evaluated by two human experts and categorized according to a scoring scheme used by McNamara, in similar applications designed for English [4]. Pupils are taught various verbalizing methods and are encouraged to use them alternately. Consequently, evaluating the explanations given by pupils is a key step in helping them improve their reading comprehension. Our evaluation criteria are centered on the knowledge used by the reader in phrasing their explanations. Starting from [4], a verbalization can be categorized as follows.

- **Paraphrase –** a restatement of the text read using other words. Paraphrasing a text forces the pupils to transform the text into a form which is more familiar to themselves. It also forces them to make a representation of the information contained within the text and to form a preliminary structure of the context; these can be considered the first steps in the understanding process of a given text;
- **Prediction** – an explanation that somehow predicts some of the information that is going to occur in the text;
- **Causally-relevant sentence –** a sentence that is closed by a causal relevant sentence of the last paragraph;
- **Pre-knowledge sentence –** an explanation in which the reader uses some previous information along with information found in the text;
- **Bridging (correlation) –** the reader links pieces of information from the text; this enables to understand how various parts are related, therefore providing a global image of the entire reading material.

If we want pupils to be assisted while reading, we are going to have one human expert taking care of a small number of them, which makes it impossible for such training techniques to be used on a large scale. Moreover, assessing the content of a verbalization is a demanding and subjectivity-laden activity, which can be assisted by computer-based techniques. These are the main motives behind the idea of using a computer program instead of, or as support for, a human tutor.

Initial experiments were conducted by McNamara and her colleagues [4]. *iSTART* can be considered the first implemented system that addresses self-explanations [10]. It has various modules that explain the SERT method to the students, one which shows them how to use those techniques using a virtual student, and another training module which asks students to read texts and give

verbalizations, evaluates them and provides an appropriate feedback. The main challenge raised by such a system is evaluating verbalizations given by pupils in accordance to the reading materials.

Therefore, the goal of our project is to enable the usage of new texts with little or no human intervention, providing fully automatic assessment as support for the human teacher. *iSTART* is dividing verbalizations into four main categories: irrelevant, paraphrases, verbalizations that use knowledge previously found in the text and verbalizations which use external knowledge from the students' experience. As stated in [5], it is easier to identify paraphrases and irrelevant explanations, but it is more difficult to identify and evaluate verbalizations which contain information coming from students' experience.

Our purpose was to create a similar program for French language and to provide support in the educational process of primary school pupils. We have integrated an evaluating module based on Latent Semantic Analysis (LSA) and a word based approach as techniques of natural language processing.

The method employed for automatically categorizing verbalizations compared them subsequently to the last paragraph read, the previous and the next ones, as shown in Fig. 1.

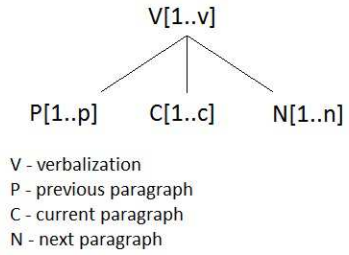

Fig. 1. Evaluation technique

The way a verbalization is included in one of the categories mentioned above is described further in Table 1. Our current research is focused only on determining what "close" and "somewhat close" mean in terms of natural language processing and on detecting verbalizations, where the paraphrasing elements prevail.

Table 1

Categorizing verbalizations logics

Verbalization type	**Text similarity**
Paraphrase	V is very close to C
Prediction	V somewhat close to N
Causally-relevant sentence	V close to the causally-relevant (hand-coded) sentences of C
Pre-knowledge	V close to a summary of the text
Bridging	P, C, V and N are very close to each other

The following sections address the architecture of our application and the role of each module in our attempt to automatically determine the nature of verbalizations provided by pupils. Section 4 presents in detail the experiments and the decisions we made based on the results of the tests regarding text similarity measurements and paraphrase detection. Section 5 comprises the conclusions and sets some research paths.

2. Architecture

The application consists of several modules (Fig. 2) and some of them address user interaction. In this section we focus mainly on the modules used for evaluating self-explanations.

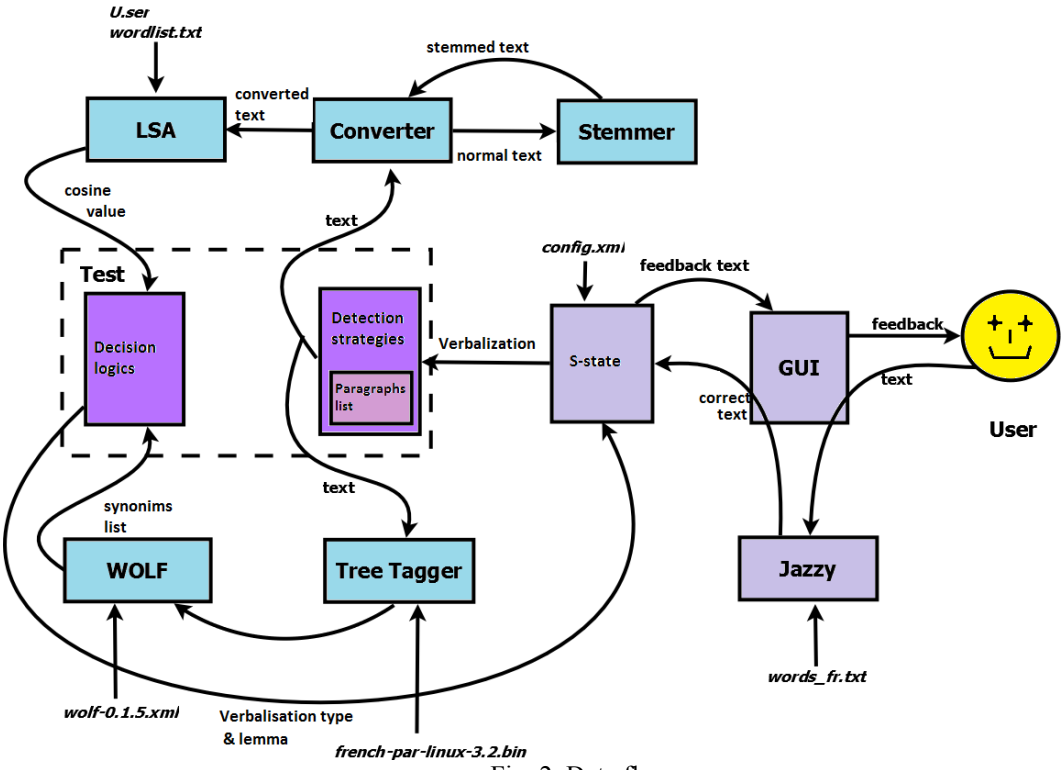

Fig. 2. Data flow

Information Flow

During initial load, the application parses a predefined configuration file and builds the state graph that is going to dictate its behavior. When a verbalization is required, as the user is typing, the words are verified by the Jazzy module (responsible for performing spellchecking) [9] and are changed with their

presumably correct form. Later on, the input is passed to the State module which requires the Test module to rate the verbalization. Depending on the rate returned by the Test module, the State module may decide to request another explanation or to move onto another paragraph.

The Test module receives the explanation in plain text and compares it with the previous, the current and the next paragraph. The similarity function is based on Latent Semantic Analysis (LSA), on one hand, and a list of important words extracted by means of Information Retrieval, on the other. Our system gets a rating from each of the two comparison methods and decides, based on experimentally determined threshold values, whether or not the two paragraphs resemble each other.

In order to perform the LSA comparison, the Test module first passes the information through the converter module, which first eliminates the punctuation, stems the entire text (only if LSA was previously trained on a stemmed corpus) and then replaces the diacritics. Afterwards, the LSA module computes the LSA vector of the paragraphs and returns the cosine similarity between the two compared paragraphs. The training corpus we used contained various texts for children. The total size of the corpus was of 6 MB consisting of plain text that had been segmented, the punctuation eliminated and the diacritics replaced. Only segments between fifty and one hundred words were kept for training.

In order to determine the resemblance between words contained in the paragraphs, the Test module has to build a list of relevant words for it. The list contains the words from the four categories recognized by WOLF and their synonyms. In order to determine their part of speech and their lemmas, the Tree Tagger is used. Each word is then looked up in WOLF and its synonyms are added to the list. Then the Tester module counts the words in the verbalization present in the relevant word list and provides a grade depending on their number of occurrences, which will be further explained in the next section. Once a decision has been made, the grade is passed to the state module, which gives the user an appropriate feedback.

3. Integrated Technologies and Approaches

This chapter addresses the main integrated technologies within our system, covering natural language processing, Latent Semantic Analysis and WOLF as a lexicalized ontology.

Spellchecking

The Jazzy module is used for the spell-checking in the user input window [9]. It uses a dictionary and tries to approximate a word using the Levenshtein distance algorithm. The dictionary is in fact a list of words and it was

obtained by parsing a French XML dictionary, Morphalou[6], which contains all the inflectional forms of French language.

Converter

The input text provided by the user is then passed to this module which is used to prepare it for the LSA analysis. The first step in the conversion is to eliminate all punctuation and to keep only the word tokens. Then, if LSA had previously been trained on a stemmed corpus, stemming is also performed on our input text. As the French diacritics had been replaced in the training corpus, we had to replace them as well in the input texts.

Stemmer

In linguistic morphology, stemming is the process for reducing inflected (or sometimes derived) words to their stem, base or root form – generally a written word form. The stem need not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root. Our actual implementation is based on Snowball [7], an open-source rule-based parser.

Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a theory and method for extracting and representing the meaning of words. Meaning is estimated using statistical computations applied to a large corpus of texts. A language corpus presents a set of constraints which the LSA is extracting in order to determine the meaning of words through concepts [5]. On a more mathematical viewpoint, LSA uses linear algebra methods, especially singular value decomposition. Since it measures similarity between words, LSA works better on specialized corpora of texts, such as texts from scientific vocabulary. In order to work fine, it is important for LSA to be applied on texts from the same domain as the one that it has been trained on.

The main strength of LSA is the power to exploit mutual constraints. Its principle is that the meaning of a paragraph can be computed as a function of the meanings of all the words it contains. Using linear algebra, each paragraph is considered a simple linear equation and a corpus a large set of simultaneous linear equations, where the variables are the occurring words. So LSA treats the corpus as a number of individual paragraphs that carry coherent meaning, converts each of them into an equation where the word represents a variable and the number of occurrences its coefficient and by solving the system, LSA computes the value of each of the words. Therefore, by using LSA it becomes possible to compute the meaning of every new paragraph using the value of its corresponding words, in a

[6] http://cnrtl.fr/lexiques/morphalou/

bag of words approach. In this method, the meaning of the words depends on the meaning of surrounding or co-occurring words. In consequence, in order to make the method efficient, we need to provide LSA with a sufficiently large learning corpus, comparable to the one that a human needs in order to acquire verbal skills. Because of the very large dimension of the equation system, the actual computations are very time and resource consuming, even on powerful, distributed computers. After performing the actual SVD decomposition, the resulted space is projected onto 300 dimensions, providing the final semantic vector space to be later used during our assessment process.

The vector of a paragraph is estimated as the sum of its components, and the similarity between paragraphs is measured in terms of cosine similarity between the two vectors. Moreover, we have included methods specific to information retrieval, more specifically Term frequency–Inverse document frequency (*Tf-IDf*), for improving the estimation of a paragraph's vector:

$$p_i = \sum_{i=0}^{k} x_i \frac{1}{f_i}(1+\log n) \tag{1}$$

where p_i is the value of the i^{th} dimension of paragraph vector, x_i is the value of the i^{th} dimension of the word's vector, n is the number of times the words appears in the paragraph, f_i is the word's frequency in the training corpus and k is the dimension of the vector space. The cosine value between two vectors is computed using the following formula:

$$\cos(\vec{x}, \vec{y}) = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2 \sum y_i^2}} \tag{2}$$

Tree Tagger

Tree Tagger[7] is a language independent part-of-speech tagger [8] and it helps to identify the four most important parts of speech recognized by WordNet: nouns, verbs, adverbs and adjectives [6]. Another important feature of this module is that it also recognizes the lemma of the word, making it easy to look for in a dictionary. In the absence of such a tool, we would had been obliged to parse and load in the memory a dictionary containing all the inflectional word forms of the French language, which would have consumed a lot of time and resources. Its main advantage is that it can work independently of the language – all it requires is a configuration file that differs from language to language.

[7] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

WOLF

WordNet is a lexical database for English [6] that groups words into sets of synonyms called *synsets*, provides short, general definitions, and records the various semantic relations between these synsets. The purpose is to produce a combination of dictionary and thesaurus that is intuitively usable, and to support automatic text analysis. It is important to specify the difference between WordNet and a thesaurus. A thesaurus groups words together based on their meaning. WordNet also interlinks groups of synonyms, therefore becoming a more powerful tool in NLP.

As the purpose of our program was to build a NLP tool for French, we had to find an alternative for WordNet available for French. The only open source reliable database available online is WOLF[8] (*WordNet Libre du Français*). It contains about thirty thousand synsets, and the sense fields are filled with the information on the sources where the lexeme was found, and not with the sense number. It is kept in an XML format, copying the syntax of the BalkaNet project. It is obvious that this project cannot match the WordNet's performances, but it is the most suitable tool we could use for French.

Similarity Measure

Test is the core module of our system that connects all the other modules and performs the computational work. It receives input from the State module and from the input files, calls the other modules in order to rate the verbalizations and returns the answer to the state module.

For implementing the *word-based heuristic*, the Tree Tagger and WOLF are used in order to create a list of relevant words for each paragraph. When a paragraph is created, the words composing it are tagged and then a list containing all the synonyms of nouns, verbs, adjectives and adverbs in the text is created. All these words are considered relevant for the text.

Later on, the words of the verbalization are tagged and, afterwards, the words in each category are counted. Then the fraction of the words in the paragraph and the words in the verbalization for each category is computed. Four fractions are obtained and a weighted average of the four is returned as an overall rating:

$$R_W = \frac{W_n \frac{n_n}{N_n} + W_v \frac{n_v}{N_v} + W_{aj} \frac{n_{aj}}{N_{aj}} + W_{av} \frac{n_{av}}{P_{av}}}{W_n + W_v + W_{aj} + W_{av}} \quad (2)$$

Where R_W is the rating returned by the function, $n_{n,}, n_v, n_{aj,}$ and $n_{av,}$ are the number of nouns, verbs, adjectives and respectively adverbs in the

[8] http://alpage.inria.fr/~sagot/wolf-en.html

verbalization which can be found in the list of relevant nouns of the paragraphs, N_n, N_v, N_{aj} and N_{av} are the length of these lists and, and W_n, W_v, W_{aj} and W_{av} are their weights in the average. All these predefined weights were determined experimentally, after running multiple iterations with incremental values.

Another separate function is used to process the *LSA similarity heuristic*. It compares each sentence of the paragraph to the entire verbalization and a weighted average of the value is computed, ignoring the two smallest values. Due to the fact that each verbalization usually contains some control sentence or sentences which are irrelevant to the comparison, and we don't want them to alter the result. The weight of an utterance is equal to the number of words it contains. The whole paragraph is also compared to the verbalization, as we know that the meaning of the paragraph as a whole can be slightly different from the meaning of each sentence individually. In this manner we cover both cases when a verbalization focuses on the whole paragraph or only on some sentences within.

Having computed these two parameters, the Test module can make a decision about the current paragraph. A lower and an upper threshold have been enforced; their values were experimentally determined. Using several texts, we observed consistent changes of thresholds, making human intervention in setting these values mandatory. If the paragraph scores low on both criteria, then we consider it to be irrelevant to the text, so that it cannot be included in any category. In a tutoring system, this type of self-explanation would lead to a request for a restatement. If both scores are higher than the upper threshold, then the verbalization is a paraphrase. Otherwise, it can be considered as being related to the paragraph, but not close enough so that it could be considered a paraphrase.

4. Experiment and Results

We have performed several tests using the combined metrics (LSA similarity and word co-occurrences approach) and were able to draw several conclusions based on our results. Our test corpora consisted in a narrative text for children (*L'étrange rencontre*, about 630-word long), divided into six paragraphs of about five sentences long, and the verbalizations for each paragraph provided by primary school pupils (from 3rd to 5th grade). Five children were asked to read the texts and to stop at predefined points and to explain what they have read up to that time. The verbalizations were then transcribed and evaluated by a human expert who identified the paraphrasing, bridging, elaboration or prediction elements in each of them, enabling us to evaluate our results. We compared the verbalizations using both our metrics with the paragraph the pupil had just read, with the paragraph preceding it and with the one following it, and we tried to decide on their nature depending on the resemblance with those paragraphs.

First we were interested in measuring the distribution of the values returned by the two employed metrics. Theoretically both return a real number between 0 and 1 which represents the degree of resemblance of two paragraphs, but we were interested in seeing what the real return range would be when real data is used. So we arbitrarily chose some of the results of comparing paragraphs with verbalizations using our two heuristics, sorted them ascending and represented them in the graphs below (Fig 2 and Fig 3).

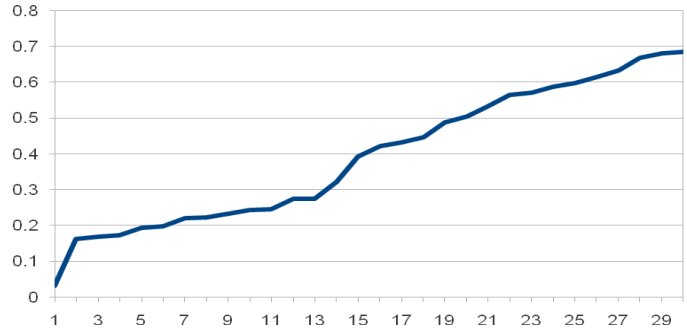

Fig. 2. Distribution of the word-based heuristic

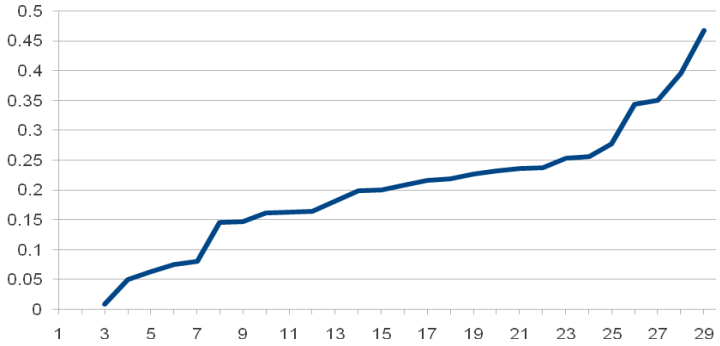

Fig. 3. Distribution of LSA-based heuristic

We can notice that the LSA has a smaller range, varying between 0 and 0.5, while the other evaluation function varies between 0 and 0.7; nevertheless, both have quite a linear evolution. This analysis helped us establish a threshold over which we could consider a verbalization to be a paraphrase.

At this point we have two metrics, both indicating the degree of resemblance of two paragraphs, but we had to decide whether the results of these two metrics are coherent or not, so we tried to evaluate the correspondence between the two metrics. Fig. 4 depicts the compared results of the two metrics on the same data. Based on these observations, we decided that the best way to

combine these two metrics was to multiply them. The combined metrics is also represented in the same chart.

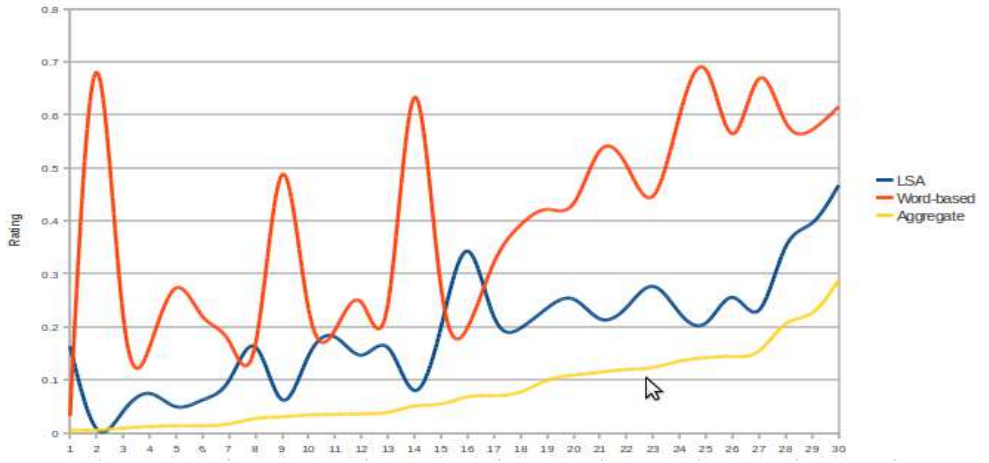

Fig. 4. Comparison between the two metrics employed

The Pearson correlation between our metrics, computed on the data in the table is 34.02%, the correlation on the LSA metric and the aggregate is 87.73%, and between the word based metric and the aggregate 68.16%. This means that the LSA metric has a bigger influence on the final grade.

Observing these results we decided to establish a threshold around 0.07 for paraphrases; the value was determined experimentally using Fig. 3 as the first value at which a significant growth can be observed. This means that a verbalization which scores more than 0.07 when compared to the corresponding paragraph can be consider a paraphrase. This threshold allowed us to identify nineteen out of the twenty seven paraphrases identified by human evaluators, which means that we were able to correctly identify 70% of the paraphrases.

At this point we were able to identify a paraphrase with a quite good precision, but we had to see if we could also identify other types of verbalizations using the data obtained from the comparisons we made. In consequence we compared the values of the current paragraph with the previous and the future ones in order to determine the similarity between verbalizations of the same type. Firstly we represented the variations of the two metrics for verbalizations containing bridging elements and for the paraphrases, in order to identify some similarities between verbalizations of the same type.

Fig. 5 shows the values returned by the word-based metrics for ten paraphrases, which represent (about one third of the total number of paraphrases of our test corpus) when compared to the previous, the current and the next paragraph. It is obvious that there is a much bigger resemblance between the

current paragraph and the verbalization, while the resemblance between it and the surrounding paragraphs is close to zero.

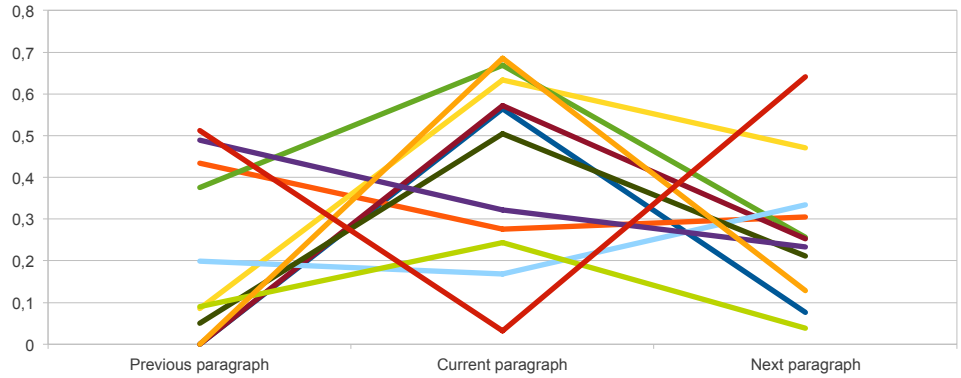

Fig. 5. Verbalizations containing paraphrases compared using the word-based heuristic

Fig. 6 shows the same thing for the LSA metrics. We notice that the graphic has the same characteristics, with a little bit more variations, which makes us conclude that the LSA method is more accurate than the other, although the average similarity values are quite low.

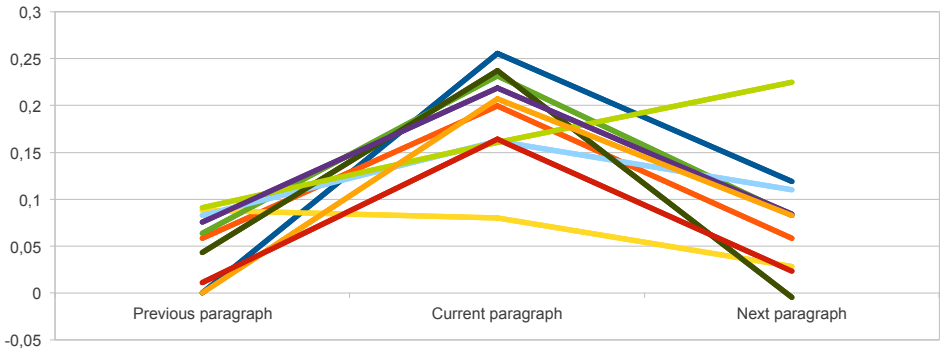

Fig. 6. Verbalizations containing paraphrases compared using LSA-based heuristic.

We made the same two graphics for verbalizations where bridging elements prevail, as it can be seen in Fig. 7 and Fig. 8.

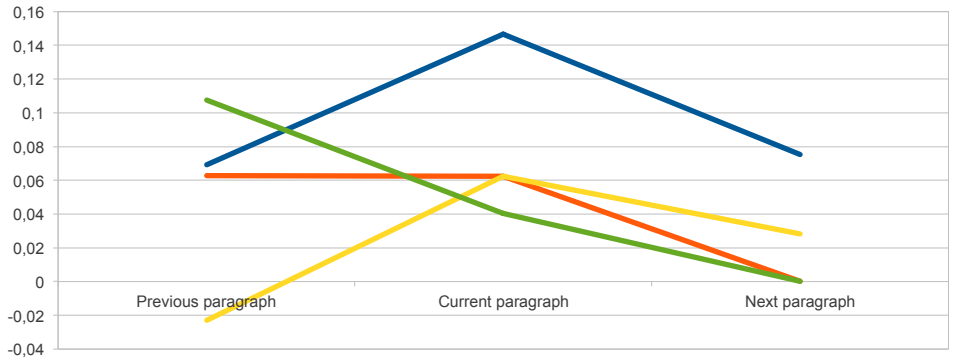

Fig. 7. Verbalizations containing bridging elements compared using LSA

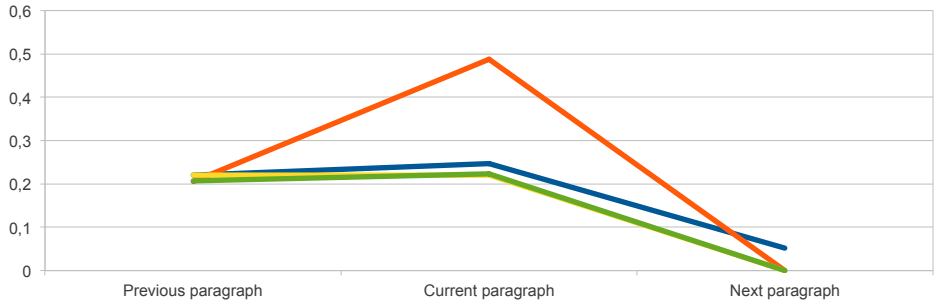

Fig. 8. Verbalizations containing bridging elements compared using word based heuristic

Even though we had only a small number of verbalizations containing bridging elements, it is obvious that no clear conclusion can be drawn by using this approach, as the results do not follow a clear pattern, so that no distinction can be made between them and a simple paraphrase.

5. Conclusions

Starting from the work of McNamara, we began to develop a natural language processing application which aims to evaluate explanations given by students during the reading process and to place it into an appropriate category, assisting the tutor in providing a customized feedback. This task is far from trivial, and requires, along with a lot of computational power, a comprehensive approach of the problem.

In order to determine the nature of self-explanations, we used LSA and a word-based heuristic to compare the verbalizations with nearby paragraphs. Our approach provided encouraging, but limited results. Therefore, we are able to identify paraphrases with quite good precision and to understand that we could not

obtain other useful information by only comparing verbalizations with some paragraphs in the text. For that we need to implement new strategies and make thoughtful use of our instruments.

Future research paths will focus on finding similarities between the verbalizations and different segments of the text in order to determine how much of the information in the text has been used by the student to explain it, all evaluated using a more formal model of discourse analysis, as shown in Table 1.

R E F E R E N C E S

[1] Chi, M.T.H., de Leeuw, N., Chiu, M.H., &LaVancher, C. (1994). *Eliciting self-explanations improves understanding*. Cognitive Science, 18, 439-477

[2] Danielle S. McNamara, Jennifer L. Scott (1999). *Training Reading Strategies*, Old Dominion University, Department of Psychology, p387-392, Norfolk USA, Erlbaum

[3] Danielle S. McNamara (2007*), Reading comprehension strategies: theories, interventions, and technologies*, 398-403, New York, Erlbaum

[4] Tenaha O'Reilly, Danielle S. McNamara, Grant P. Sinclair, (2004)*, iSTART: a web-based reading strategy a Intervention that improves students' science comprehension*, University of Memphis

[5] Simon Dennis, Walter Kintsch, Thomas K. Landauer, Danielle S. McNamara. *Handbook of latent semantic analysis*, 2007.

[6] Christiane Fellbaum (1998, ed.) *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press.

[7] M.F. Porter (2001) - Snowball: *A language for stemming algorithms*, available online at <http://snowball.tartarus.org/texts/introduction.html>

[8] Helmut Schmid (1994): Probabilistic Part-of-Speech Tagging Using Decision Trees. *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UK, available online at <ftp://ftp.ims.uni-stuttgart.de/pub/corpora/tree-tagger1.pdf>

[9] Mindaugas Idzelis, *The Java Open Source Spell Checker*, available online at <http://jazzy.sourceforge.net/>

[10] G. Jackson, R. Guess, D. McNamara*, Assessing Cognitively Complex Strategy Use in an Untrained Domain*, University of Memphis, 2009