**WANNACHAI KAMPEERA**
**SYLVIANE CARDEY**
**Centre de Recherche en Linguistique et Traitement Automatique des Langues**
**Université de Franche-Comté**
**Besançon, France**
**wannachai.kampeera@univ-fcomte.fr, sylviane.cardey@univ-fcomte.fr**

# *Paraphrases in Natural Language Processing*

## 1 Introduction

It is crucial to define the term 'paraphrase' before moving to other related issues. The definition of 'paraphrase' has been given differently according to authors (depending on linguistic levels: words, phrase, sentence, text; different linguistic approaches: linguistic paraphrases[1], pragmatic paraphrases[2], etc.). In this paper, we define the term 'paraphrase', and eventually its plural form 'paraphrases' as expressions which are expressed differently, but have approximately the same meaning. For instance, the expression "John appreciates French food" is a paraphrase of the expression "John is fond of French cuisine" or "John likes French cooking". They are said to be 'paraphrases' of each other. Besides, the term 'paraphrase' also refers to a semantic (quasi-) equivalence relation over expressions.

From the dynamic point of view the term 'paraphrasing' refers to the process which consists in rephrasing an expression $E_1$ into other expressions $E_2$, $E_3$, $E_n$, which keep more or less the same meaning.

The paraphrase processing module has become an essential component in a large number of natural language applications. For example, the verbatim[3] classifier 'Classificatim' (Cardey et al., 2006) relies on complex linguistic paraphrasing rules (or synonym rules) to tag and classify verbatims which address the same subject. Similarly, in information extraction, paraphrase processing is applied to retrieve further relevant information (Stevenson et al., 2005; Shinyama et al., 2003). In question-answering systems, a paraphrase extraction/generation module improves significantly the system's performance (Duboue & Chu-Carrol 2006; Duclaye et al., 2003). In multi-documents summarization, the fusion of similar information over documents is made by a paraphrase recognition module in order to avoid redundancies (Barzilay et al., 1999). Paraphrase processing has been shown to be useful in many other language applications: machine translation (Koehn, 2009), natural language generation (Reiter and Dale, 2000).

In section 2, we discuss the linguistic background of paraphrases. We present a linguistic theory for which a paraphrasing system has been developed. We provide in section 3 a review of three major tasks in paraphrase processing as well as current prevailing methodologies. We also discuss the advantages and drawbacks of these methods. We present in section 4 our on-going research project concerning a multi-role paraphrases generator.

## 2 Paraphrases in Linguistic Analysis

### *2.1 Paraphrasing system in Meaning-Text Theory*

The linguistic phenomenon of paraphrasing is fundamental in linguistic analysis. In Meaning-Text theory (Mel'čuk, 1997), the mechanism of paraphrasing can be compared to a pivot between meaning and text. According to this theory, a 'meaning' (or a semantic representation) corresponds to the meaning of all of the paraphrases (texts having the same meaning), and these paraphrases are obtained by applying paraphrasing rules. In this paper, we will only focus on the paraphrasing system (Milićević, 2007), an interesting formal linguistic framework for paraphrasing rules.

The paraphrasing system is primarily studied under the Meaning-Text theory. This linguistic framework provides linguistic paraphrasing rules at different linguistic levels: semantic, lexical-syntactic, syntactic, and morphological. The most interesting level for natural language processing seems to be lexical-syntactic paraphrasing rules. In fact, such lexical-syntactic paraphrasing rules are, on the one hand, powerful enough to generate sophisticated and complex paraphrases. On the other hand, they are more feasible regarding their implementability, this in comparison with semantic paraphrasing rules which require essentially a standardization of lexical entries' definitions (Milićević, 2007, pp. 165-166). In other words, the machine needs a standardized-unambiguous definition of words so that it can perform accurate processing on paraphrases. Currently, methodologies for standardized lexical entries' definitions are under development.

Although the lexical-syntactic paraphrasing rules above were proposed under the Meaning-Text theory, it is, in general, possible to adapt such linguistic rules to rule-based language processing systems given that such rules have been written under a formal representation scheme. Some references for the implementation of these rules can be found in (Milićević, 2007, p.150). It should be noted that paraphrasing rules formalized in this manner are not exhaustive.

[1] The speaker can decide whether two expressions convey the same information by using solely his linguistic knowledge, without encyclopedic knowledge, nor context.
[2] To see the paraphrase relation between two expressions, the speaker needs the communication situation, or he/she has to use his/her world knowledge, or other skills apart from linguistic knowledge.
[3] In our context, verbatims include all types of consumers' message. A verbatim can be an email, a letter, a transcription of telephone calls, etc.

## 3 Methodologies for Paraphrase Processing from the Linguistic Point of View

A possible typology for the paraphrase processing task is to divide the task into three major sub-tasks: generation, recognition (identification), and extraction (acquisition). Paraphrase generation systems take an expression as input and produce its paraphrases as output. Paraphrase recognition consists in deciding whether two given expressions are paraphrases of each other (whether their meaning is approximately equal enough according to some established threshold). The paraphrase extraction task does not need specific input; the latter can be a text, a monolingual as well as bilingual corpus. However, there can be some 'seeds' (keywords or key-patterns) which serve as launchers for an identifying process in the systems deploying bootstrapping methods, as in (Barzilay and McKeown, 2001). The objective of a paraphrase extractor is to discover the paraphrases in a given text, and then put them in an equivalent expressions table.

### *3.1 General tendencies for methodologies used in paraphrase processing*

In trying to reduce time and financial outlay to a minimum, with maximal automation and productivity, many language processing applications have recently been developed using statistical approaches with large corpora, or hybrid approaches. When it comes to extracting (linguistic) rules, machine learning techniques can become, in some cases, a substitute for linguists. Likewise, many researchers in the paraphrase processing field apply hybrid methods which rely on statistical calculations involving linguistic features.

### *3.2 Recognition*

In paraphrase recognition competitions applied to the Microsoft Research Paraphrase Corpus[4], the best score[5] with 76.6 % of accuracy was achieved by (Kozareva and Montoyo, 2006), who applied lexical overlap (counting common words in two texts), lexical similarity and semantic similarity measures. The 'similarity measures' are a means to measure how close two expressions are by applying certain mathematic formulae. If the established threshold is attained, the two expressions are paraphrases. Kozareva and Montoyo used Wordnet[6] to extract semantic information, e.g. synonym relations and semantic relations between nouns and verbs. They combine several similarity measures, thus several outputs are given. Finally, they use voting algorithms to select the best answer. Measuring the similarity at different linguistic levels, i.e. lexical, semantic, and syntactic, is a widespread approach used by paraphrase recognition tasks. Some other competitive extraction systems applying similar methods are (Fernando and Stevenson, 2008; Corley and Mihalcea, 2005).

It may be surprising that shallow lexical similarity measures (n-grams with vector-space model, edit-distance) offer competitive results, these even better than some complex rule-based systems; see (Dagan et al., 2007) in a tutorial on textual entailment (entailed paraphrases). Validating the paraphrase relation between two expressions by counting common words or letters seems random. In fact two graphically similar expressions do not convey necessarily the same information, e.g. "My dog likes children" vs. "My dog bites children". In the same way, two different expressions can have the same meaning, e.g. "Graham Bell invented the first practical telephone" vs. "The earliest operational phone was created by Graham Bell". Shallow lexical recognition will probably fail in such cases.

Nevertheless, natural language is complex in real world contexts. On the one hand, while a rule-based system is highly reliable for language and domain specific tasks, this is questionable faced with how to provide coverage by means of linguistic paraphrasing rules which can change according to domains, languages, and the size of the data. On the other hand, shallow lexical methods can naively fail in some contexts (as in the previous two examples), but they operate in the same way whatever the languages, the domains, or the sizes of data are (only the lexical similarity counts). Furthermore, when similarity measures make use of linguistic knowledge, e.g. Wordnet, semantic and syntactic relation, better performance can be obtained. In any case, we remark that so far the best system combining several similarity measures performs with an accuracy of only 76.6%. This would suggest that there is still a lot of progress to be made and in the future better methods for paraphrases recognition should be found.

### *3.3 Extraction*

In paraphrase extraction, prevailing methods are: syntactic-based/graph-based, pivot-based techniques, and bootstrapping. Syntactic-based techniques consist in parsing a text to obtain parse trees, e.g. dependency trees X <-- like --> Y. Next, the system looks for expressions which contain the nodes X and Y, such as X <-- is_fond_of --> Y. If the matching frequency attains a certain threshold, these syntactic paths are paraphrases. This technique takes its inspiration from the distributional hypothesis (Harris, 1964) which states that elements occurring in the same linguistic environment tend to have similar meanings. We can take as an example for this technique based on representation and matching the systems conceived by (Lin and Pantel, 2001; Ibrahim and al., 2003; Barzilay and Lee, 2003), and (Pang and al., 2003) with a lattice for the text representation instead of a syntactic path.

The pivot approach borrows its idea from statistical machine translation. (Bannard and Callison-Burch, 2005; Madnani and al., 2007; Zhao and al., 2008) make use of parallel bilingual corpora which is precisely a table of aligned expressions between source language L1 and target language L2. Let L1 be English, L2 French. To extract the paraphrases of the term 'like' in L1, we look at its correspondent in L2, we find 'aimer'. Now, from 'aimer' in L2, we look back to its correspondents in L1, we find 'like', 'appreciate', 'is fond of', which are

[4] http://research.microsoft.com/en-us/downloads/607d14d9-20cd-47e3-85bc-a2f65cd28042/default.aspx
[5] http://www.aclweb.org/aclwiki/index.php?title=Paraphrase_Identification_(State_of_the_art)
60
[6] http://wordnet.princeton.edu/

paraphrase of each other. According to (Bannard and Callison-Burch, 2005), the pivot approach works especially well at word or phrase level; however there are often errors concerning automatic alignment.

In bootstrapping, seeds, often nouns or verbs, are launched to retrieve their contexts. Then, the contexts are, in turn, used to find more seeds and so on, until the system does not find anything new. Let us take a concrete example in (Szpektor et al., 2004) to show how bootstrapping works. The seed 'prevents' is launched. The system finds the pattern 'X prevents Y'. Next, the pattern 'X_Y' is used to find other seeds: 'X provides protection against Y', 'X decreases the risk of Y', etc. The system renews the process using 'provides protection against' and 'decreases the risk of' as seeds, and so on. In the end, 'prevents' has its paraphrases: 'provides protection against', 'decreases the risk of', etc. This technique, once again, is based on distributional hypothesis which seems too simplistic for the paraphrase extraction task.

Indeed, it is possible that two expressions occurring in the same linguistic environment convey the same information. However, these two expressions found in the same context can also be antonyms, or indeed anything. For example, in a given coherent text, there can be the sentence "X hated Y at first sight" at the beginning of the text. Later, there can be the sentence "After having worked together for a while X fell in love with Y" and also "X liked talking about Y". Basing on distributional hypothesis to extract the "same meaning" appears to be too risky. Yet, it is certainly a well-founded theory for a "same part-of-speech" tagger approach. Besides, bootstrapping and graph-based techniques require a good resource of paraphrases: large parallel or comparable monolingual corpora which are fewer compared to existing parallel bilingual ones.

### *3.4 Generation*

Contrary to extraction and recognition, the number of publications on paraphrases generation is small (Androutsopoulos and Malakasiotis, 2009). The reason for this difference is that there are neither established benchmarks nor competitions for paraphrase generation. In fact, for a given input expression, a generation system can output its paraphrases as much as it can. As a result, it is impossible to establish benchmarks fixing, for example, the number of generated paraphrases. Also, the question of how to judge the quality of generated paraphrases by different systems remains problematic.

However, current methods used in extraction and those in generation are similar. As in extraction, we find bootstrapping (Duclaye and al., 2003) and pivot-based techniques (Quirk and al., 2004; Duboue and Chu-Carroll, 2006, with several translation engines) which are gaining in popularity. We remark that pivot-based techniques can suffer from error accumulation caused by each transfer process.

It is quite surprising to see much of the work on paraphrase generation moving towards statistical corpus-based methods. In fact, contrary to paraphrases recognition where the domination of statistical approaches can be easily justified, the paraphrase generation task should be handled efficiently by rule-based systems. As the input to the generator is an expression or any text representation at a certain linguistic level, the next processing step applies linguistic paraphrasing rules together with controlling or filtering modules on the generated output. Rule-based paraphrase generators would appear to be able to provide sufficiently sophisticated and varied paraphrases because both the input and the rules to apply are known (this is not the case for recognition and extraction). Given that our on-going research is directly concerned with paraphrases generation, we discuss this issue in more detail in the next section.

## 4 The On-going Research Project: a Rule-based Model for Paraphrases Generation

Before describing the characteristics of our own paraphrase generation model, we summarize the general features of a paraphrase generation system. A paraphrase generator takes an expression as input, and outputs its paraphrases. Unlike extraction and recognition, it is, so far, impossible to establish benchmarks for paraphrase generators. The reason for this impasse is that the number of output expressions for a given input cannot be determined. Also, the question of the paraphrases' quality remains unsolved. In fact, judging the grammaticality of an expression is feasible but evaluating the quality of the expressions output by different systems remains impractical. A likely solution is to resort to human intervention for the quality evaluation phase. Again, new questions arise: How many persons do we need? How long will it take to evaluate thousands of outputs per system? There will be without doubt disagreement among evaluators; so how to deal with this? In brief, there is no 'the best paraphrase generator' criterion for the time being.

In spite of those issues, we have defined the characteristics of our paraphrase generation model as follows (note however that this model is for generators aiming at quality, and not quantity which may be the objective of certain systems):

- ✓ Rule-based: it should be rule-based in order to take the maximum of the benefits from available linguistic knowledge on paraphrasing. We have two main sources for paraphrasing rules. The first one is the corpus itself from which we are extracting paraphrasing rules. This corpus from a food processing industry is the same one used by the Classificatim system (Cardey et al., 2006). It provided an ideal source for paraphrasing rules given that it is made up of vertabims from a very large number of consumers having more or less the same purposes but expressing them in various ways. The other source is a set of available paraphrasing rules described in linguistic theories. Theoretical descriptions will serve as tools to position the most general rules in relation to the specific ones observed in the corpus.
- ✓ Good control on generative power: it should produce a set of sophisticated and varied paraphrases, which is not excessive but responds to the need, rather than generating an enormous quantity of paraphrases containing also many poor ones. We will need to discard the rules which are too specific, e.g. paraphrasing

rules which are valid in a very limited context. Accordingly, we will take into consideration only those which are relevant in most cases.

- ✓ Flexible: the model should allow paraphrases by different linguistic means, e.g. lexical paraphrases, syntactic paraphrases, lexical-syntactic paraphrases, etc. The system will be able to switch from a generation mode to another, using one mode or combining them. This feature provides a possibility to stylize expressions according to the user's need.
- ✓ Unambiguous: like most natural language applications, a paraphrase generator should minimize ambiguity.
- ✓ Possibly bi-directional: as the model is rule-based, it should also be able to play the role of a paraphrase recogniser. This is in fact the inverse process of generation. However, instead of limiting the number of paraphrasing rules as in generation phase, the recognition process must contain as many paraphrasing rules as possible. Consequently, the model will be a common framework for the paraphrase generation and recognition task.
- ✓ General and transferable: the design of the model will be as general as possible. Although we are initially working on a domain and language specific corpus (French), this is to be considered as a case specific study from which we can conceive the model in trying to bring linguistic descriptions to the most general level. In this way, the transferability of the model to other domains and languages should become possible.

We are developing a rule-based paraphrase generation model which will have the characteristics described above. This work is inspired by the verbatim classifier Classificatim, more precisely, by paraphrase (synonym) rules in this system. The Classificatim system is rule-based and it achieved an 84% success rate in classifying verbatims and a 99% success rate after automatic verbatim normalisation. It is evident that such projects require time and in-depth linguistic investigation, but the final products offer high performance, easy maintenance and extension, and especially so, reliable results. This is the reason for which we direct our paraphrase generation model towards deep linguistic analysis.

## 5 Conclusion

We have presented a brief state-of-the-art review on the research in paraphrase processing. We have also provided some remarks from the linguistic point of view for each paraphrases processing task. We have underlined the importance of the research on paraphrases with respect to both natural language applications and a linguistic theoretical model. We have introduced a linguistic theory in which paraphrasing rules are formally described; this can be an interesting resource for paraphrasing rules and paraphrasing mechanisms. We have outlined, in the last section, our research project on a rule-based paraphrase generation model and its main features. The paraphrase processing module is crucial in a great number of language applications. Various research directions on computational techniques, as well as linguistic descriptions for paraphrases, are open to us.

## References


Androutsopoulos, I., & Malakasiotis, P. (2009). A Survey of Paraphrasing and Textual Entailment Methods. Journal of Natural Language Processing 11, pp.151-198.

Bannard, C., & Callison-Burch, C. (2005). Paraphrasing with bilingual parallel corpora. The 43rd Annual Meeting on Association for Computational Linguistics Ann Arbor, pp. 597-604, Michigan

Barzilay, R., & Lee, L. (2003). Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In Proc. of the Human Language Technology Conf. of NAACL, pp. 16–23, Edmonton, Canada.

Barzilay, R., & McKeown, K. (2001). Extracting paraphrases from a parallel corpus. The 39th Annual Meeting of ACL, pp. 50-57, Toulouse, France.

Barzilay, R., McKeown, K., & Elhadad, Michael. (1999). Information Fusion in the Context of Multi-Document Summarization. The 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics College Park, pp.550-557, Maryland, USA.

Cardey, S., Gentilhomme, S., Greenfield, P., Bioud, M., Dziadkiewicz, H., Kuroda, K., Marcelino, I., Melian, C., Morgadihno, H., Robardet, G., & Vienney, S. (2006). The Classificatim Sense-Mining System, in Advances in Natural Language Processing, coll. Lecture Notes in Artificial Intelligence, Vol. 4139, Springer-Verlag, pp. 674-684, ISBN 3-540-37334-9.

Corley, C., & Mihalcea, R. (2005). Measuring the Semantic Similarity of Texts. Workshop on Empirical Modeling of Semantic Equivalence and Entailment Ann Arbor, USA. Association for Computational Linguistics, pp 13-18.

Dagan, I., Roth, D., & Zanzotto, F.M. (2007). Tutorial on textual entailment. Prague: ACL.

Duboue, P., & Chu-Carrol, J. (2006). Answering the question you wish they had asked: the impact of paraphrasing for question answering. The Human Language Technology Conference of the NAACL New York, pp.33-36. USA. Association for Computational Linguistics.

Duclaye, F., Yvon, F., & Collin, O. (2003). Learning paraphrases to improve a question-answering system. In Proc. of the EACL Workshop on Natural Language Processing for Question Answering, pp. 35–41, Budapest, Hungary.

Fernando, S., & Stevenson, M. (2008). A semantic similarity approach to paraphrase detection, Computational Linguistics UK (CLUK 2008) 11th Annual Research Colloquium.

Harris, Z.S. (1964). Distributional Structure. In Katz, J., & Fodor, J. (Eds.). The Philosophy of Linguistics, pp. 33–49. Oxford University Press.

Ibrahim, A., Katz, B., & Lin, J. (2003). Extracting structural paraphrases from aligned monolingual corpora. In Proc. of the ACL Workshop on Paraphrasing, pp. 57–64, Sapporo, Japan.

Koehn, P. (2009). Statistical Machine Translation. Cambridge University Press.

Kozareva, Z., & Montoyo, A. (2006). Paraphrase identification on the basis of supervised machine learning techniques. Advances in Natural Language Processing: 5th International Conference on NLP (FinTAL 2006) Turku, Finland, pp. 524-533.

Lin, D., & Pantel, P. (2001). Discovery of inference rules for question answering. Natural Language Engineering vol 7. Isuue 4, pp. 343-360.

Madnani, N., Ayan, F., Resnik, P., & Dorr, B. J. (2007). Using paraphrases for parameter tuning in statistical machine translation. In Proc. of 2nd Workshop on Statistical Machine Translation, pp. 120–127, Prague, Czech Republic.

Mel'čuk, I. (1988). Paraphrase et lexique dans la théorie linguistique sens-texte, dans Lexique et paraphrase. Lexique Ed. Gabriel G. BES, Catherine FUCHS Villeneuve d'Ascq : Presses universitaires de Lille, pp. 13-54. ISBN 2-907170-00-7

Milićević, J. (2007). La paraphrase, Modélisation de la paraphrase langagière. Sciences pour la communication Ed. Bern, Berlin, Bruxelles, Frankfurt am Main, New York, Oxford, Wien, ISBN 978-3-03911-197-8

Pang, B., Knight, K., & Marcu, D. (2003). Syntax-based alignment of multiple translations: extracting paraphrases and generating new sentences. In Proc. of the Human Lang. Techn. Conf. of NAACL, pp. 102–109, Edmonton, Canada

Quirk, C., Brockett, C., & Dolan, W. B. (2004). Monolingual machine translation for paraphrase generation. In Proc. Of the Conf. on Empirical Methods in Natural Language Processing, pp. 142–149, Barcelona, Spain.

Reiter, E., & Dale, R. (2000). Building Natural Language Generation Systems. Cambridge University Press.

Shinyama, Y., & Sekine, S. (2003). Paraphrase acquisition for information extraction. In Proc. of the ACL Workshop on Paraphrasing, Sapporo, Japan.

Stevenson, M., & Greenwood, M. A. (2005). A Semantic Approach to IE Pattern Induction. In ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 379–386, Morristown, NJ, USA. Association for Computational Linguistics.

Szpektor, I., Tanev, H., Dagan, I., & Coppola, B. (2004). Scaling Web-based acquisition of entailment relations. In Proc.of the Conf. on Empirical Methods in Natural Language Processing, Barcelona, Spain.

Zhao, S., Wang, H., Liu, T., & Li, S. (2008). Pivot approach for extracting paraphrase patterns from bilingual corpora. In Proc. of the 46th Annual Meeting of ACL: Human Language Technologies, pp. 780–788, Columbus, OH.