Evaluation and Classification of Syntax Usage in Determining Short-Text Semantic Similarity

Vuk Batanović and Dragan Bojić

Abstract — **This paper outlines and categorizes ways of using syntactic information in a number of algorithms for determining the semantic similarity of short texts. We consider the use of word order information, part-of-speech tagging, parsing and semantic role labeling. We analyze and evaluate the effects of syntax usage on algorithm performance by utilizing the results of a paraphrase detection test on the Microsoft Research Paraphrase Corpus. We also propose a new classification of algorithms based on their applicability to languages with scarce natural language processing tools.**

Keywords — **natural language processing, MSRPC, parsing, part-of-speech tagging, semantic role labeling, short-text semantic similarity, syntax, word order.**

I. Introduction

DETERMINING the semantic similarity of short texts is a process in which a value is assigned to the given texts according to the level of semantic relatedness between them. Short-text semantic similarity (STSS) systems generally provide a score between zero and one, where zero represents complete semantic dissimilitude, and one total semantic equivalence. The semantic similarity of short texts is especially noteworthy since short texts are widely used today in the form of search engine queries and results, news headlines and snippets, comments on various social networks, etc.

There are several natural language processing (NLP) problems which depend upon the usage of some measure of text semantic similarity. Such problems include text summarization and classification, question answering, information retrieval, etc. In text summarization the choice of sentences to be included in the summary is crucial. During that process, particularly when creating a summary based on multiple documents, it is important to avoid picking a sentence that contains the same information as one of the already chosen sentences [1]. In information retrieval or question answering systems, it is often the case that the query consists of information worded in a manner different from the one used in the document which holds the answer. By taking these variations into account significant performance improvements can be achieved [2].

There are two main ways of determining semantic similarity: the *statistical* approach, which is based on using text corpora, and the *topological* approach, based on using expert knowledge. The statistical method relies on the distributional hypothesis according to which words with similar meanings tend to appear in similar contexts [3]. By applying this hypothesis to a large text corpus, it is possible to create a semantic space which specifies how many times each word appeared in each context. A context is usually either some other word in whose proximity the observed word appeared or a document. Through this process each word is effectively assigned its own context vector which makes it possible to compare word meanings by comparing their respective context vectors. The main advantage of the statistical method lies in the fact that the only resource required to create a semantic space is an unannotated text corpus in the desired language.

In the topological approach, the level of semantic similarity between two words is determined by using man-made knowledge bases, e.g. the *WordNet* [4]. Since these resources are created by using expert human knowledge, they are able to model the degrees of semantic relatedness quite successfully, when combined with suitable distance metrics. Their main drawback is the significant labor required in order to construct them.

The remainder of this paper is structured as follows: we first give an overview of the different types of syntactic information that STSS systems can utilize, as well as the tools used to obtain them. We then outline various STSS algorithms and the way each one uses syntactic information. After that we present an evaluation of said algorithms and their characteristics and we propose a new algorithm classification based on their applicability to languages with scarce NLP tools. Finally, we suggest some possible avenues of future research.

II. Syntactic Information and Syntax-Processing Tools

The most basic kind of syntactic information obtainable from a text is the order in which words appear in it. No language-specific tools are required in order to utilize word order data, making this type of information easily accessible in all situations.

Paper received February 24, 2014; revised April 3, 2014; accepted April 23, 2014. Date of publication July 31, 2014. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jovan Đorđević.

This paper is a revised and expanded version of the paper presented at the 21th Telecommunications Forum TELFOR 2013.

The work was partially supported by the TR32047 research project of the Ministry of Education, Science and Technological Development of the Republic of Serbia.

Vuk Batanović is with the School of Electrical Engineering, University of Belgrade, Serbia (e-mail: bv115045p@student.etf.rs).

Dragan Bojić is with the School of Electrical Engineering, University of Belgrade, Serbia (e-mail: bojic@etf.rs).

The most frequently used type of syntactic information is part-of-speech (POS) tags, generated by language-specific part-of-speech taggers (e.g. the English language POS taggers are typically able to distinguish between 36 different parts of speech defined in the Penn Treebank Project [5]). The proper assignment of a POS tag represents a classification problem, which is why POS taggers are usually created by applying a supervised machine learning method on a text corpus that had previously been hand-annotated with the correct POS tags. Modern POS taggers achieve very high accuracies (around 97%) [6], making them an indispensable language tool utilized by all advanced syntactic analysis techniques.

Shallow parsing or chunking is a process which identifies the constituents within a sentence, e.g. noun groups, verbs, verb groups, etc. However, shallow parsing does not specify the internal structure of these constituents, nor does it determine their role in the sentence. Full parsers, in contrast, do generate a representation of sentence structure, typically in the form of either constituency-based or dependency-based parse trees. Similar to POS taggers, most modern parsers are statistical i.e. they employ machine learning techniques in conjunction with a training corpus of hand-parsed data. This is the reason parsers for different languages have to be created separately, which is an even greater undertaking than the construction of a POS tagger since the manual parsing of sentences is a slow and arduous process. The best statistical parsers attain good accuracy levels of around 91% [7], but they still generate more errors than POS taggers.

The most advanced method of obtaining syntactic information is the syntactic–semantic analysis called Semantic Role Labeling (SRL). SRL can be viewed as a more complex form of parsing in which labels are assigned to constituents according to their roles in the sentence. Hence, SRL modules are able to deduce which words represent the subject, the object, the main verb, and so on. The appeal of SRL lies with the very high level of information being generated. Nevertheless, state-of-the-art SRL performance remains noticeably lower than that of parsers and POS taggers, ranging between 70% and 90%, depending on the used training/testing corpus and the evaluation procedure [8].

III. Statistical Similarity Algorithms

Mihalcea et al. [9] proposed a general basic STSS method on which they tested several word-to-word similarity algorithms. Two of these utilize statistical techniques: PMI-IR (*Pointwise Mutual Information – Information Retrieval*) and LSA (*Latent Semantic Analysis*). In the case of PMI-IR Mihalcea et al. used the contents of the AltaVista search engine index as the training corpus, whereas for LSA they utilized the British National Corpus containing around 100 million words. Their method compares texts in a bag-of-words manner by finding the most similar word in the second text for each word from the first and vice versa. During this process, the system employs POS tags to prevent words with different parts of speech from being paired up.

Islam and Inkpen [10] also devised their method around the bag-of-words principle, but they included a string similarity metric, and opted for a different variant of the PMI algorithm, called SOC-PMI (*Second Order Co-occurrence PMI*), which was applied on the British National Corpus. They experimented with basic syntactic information in the form of the common-word order in the two texts. Furlan et al. [11] considered a modified version of the Islam and Inkpen approach in which the COALS (*Correlated Occurrence Analogue to Lexical Semantics*) algorithm, trained on a corpus of article abstracts from the English-language Wikipedia, is used instead of SOC-PMI.

Lintean and Rus [12] proposed a bag-of-words approach based on greedy word pairing. In it, the LSA algorithm is trained on the TASA corpus containing over 10 million words. Their method uses POS tags in a similar vein as [9], but they also considered a weaker form of this restriction in which they utilized only basic word classes, such as verbs, instead of the more specific categories like infinitives, participles, past tenses, etc.

Blacoe and Lapata [13] created a distributional memory (DM) model that relies on weighted word-link-word tuples arranged into a third-order tensor. Different matrices can be extracted from such a tensor, creating different semantic spaces suitable for different problems. A POS tagger and a dependency parser are required for the construction of this model. Blacoe and Lapata combined several large corpora, including the British National Corpus and the English-language version of Wikipedia, totaling over three billion words. In addition, they considered a simpler approach called SDS (*Simple Distributional Semantic Space*) that was created through a direct application of distributional semantics on the British National Corpus. Unlike the DM model, SDS does not employ any syntactic information or text preprocessing.

IV. Topological Similarity Algorithms

Mihalcea et al. [9] also tested their STSS approach in conjunction with six algorithms that depend on the WordNet as a knowledge base. These algorithms differ amongst themselves by the specific metric used to determine the word-to-word semantic distance. Moreover, Mihalcea et al. combined all topological and statistical similarity measures and evaluated their joint performance.

Fernando and Stevenson [14] examined the same six WordNet metrics on a bag-of-words model in which all word-to-word similarities are taken into account. Since the majority of WordNet measures are only able to compare the meanings of words belonging to the same word class, a POS tagger was utilized in a manner similar to [9].

In addition to the variant which utilizes the LSA algorithm, Lintean and Rus [12] also experimented with five WordNet metrics on the same model, making use of POS information in the already described manner.

Liu et al. [15] formulated a method which combines a semantic distance measure based on WordNet knowledge with a correlation coefficient of word order similarity. This coefficient is calculated between the original and relative word-index vectors of given sentences.

Ramage et al. [16] introduced an STSS model in which

TABLE 1: AN OVERVIEW AND CLASSIFICATION OF STSS ALGORITHMS ACCORDING TO THEIR SYNTAX USAGE

Algorithm	Accuracy	Precision	Recall	F-measure	Syntactic information/tools				Type
					Word order	POS tags	Parsing	SR L	
Statistical similarity									
Mihalcea et al. – LSA	68.40%	69.70%	95.20%	80.50%		+			W
Mihalcea et al. – PMI-IR	69.90%	70.20%	95.20%	81.00%		+			W
Furlan et al. – COALS	70.32%	/	/	/					D
Islam and Inkpen	72.64%	74.70%	89.10%	81.30%	+				D
Lintean and Rus – LSA	73.00%	77.30%	84.00%	80.50%		+			W
Blacoe and Lapata – SDS	73.04%	/	/	82.33%					D
Blacoe and Lapata – DM	73.51%	/	/	82.16%		+	+		L
Topological similarity									
Lee et al.	/	75.30%	55.60%	63.90%			+	+	L
Furlan et al. – ConceptNet	68.23%	/	/	/				+	L
Mihalcea et al. – Combined	70.30%	69.60%	97.70%	81.30%		+			W
Ramage et al.	70.80%	/	/	80.10%		+			W
Li et al.	70.80%	70.30%	97.40%	81.60%			+		L
Oliva et al. – VECTOR	70.82%	74.15%	90.32%	81.44%		+	+	+	L
Oliva et al. – JCN	70.87%	74.47%	84.17%	79.02%		+	+	+	L
Liu et al.	73.60%	74.50%	91.60%	82.20%	+				D
Fernando and Stevenson	74.10%	75.20%	91.30%	82.40%		+			W
Lintean and Rus – LCH	75.70%	78.30%	87.90%	82.80%		+			W

two bags-of-words are not compared directly. What is compared instead are the distributions induced by each text when used as a seed of a random walk over a graph created by using WordNet and corpus statistics. POS tags are utilized both an as integral part of the graph construction process and in determining the initial distribution over the state space for a particular given short text.

Li et al. [17] designed their approach around a shallow parsing method which breaks sentences down into noun, verb and preposition phrases. The comparison of sentence semantics is performed by comparing the meanings between their respective constituents and then combining the similarities of the three kinds of phrases. Individual word meanings are obtained from the WordNet, using the same approach as the one in [15].

Lee et al. [18] created a syntax-processing mechanism that relies on a list of typed dependencies produced by a parser. Their method converts this list into a set of subject-verb-object syntactic patterns, thereby performing basic semantic role labeling. These sets are then compared between the given short texts by comparing the appropriate parts of each pattern pair. This process is performed using not only a WordNet-based semantic measure, but a string similarity metric as well.

Furlan et al. [11] proposed an algorithm which relies on the *ConceptNet* semantic network as its knowledge base. They employed an SRL mechanism able to extract subject-verb-object tuples from a given sentence.

Oliva et al. [19] considered several WordNet metrics for calculating word-to-word similarity and for the majority of them they utilized a POS tagger for the same reason as [9] and [14]. However, the *SyMSS* model of Oliva et al. is significantly more complex since it performs a deep joint dependency-syntactic and semantic analysis of the given texts. Once the syntactic information is processed the model pairs up the words that have the same syntactic function and compares them on a semantic level. SyMSS also lowers the final similarity score in cases where one sentence contains certain syntactic structures not present in the other, i.e. when an indirect object appears in only one sentence and not the other.

V. EVALUATION AND CLASSIFICATION

The results of all described algorithms, sorted according to their accuracy levels, are displayed in Table 1. In cases where multiple variations of a basic approach are offered, the best-performing options were selected. In addition, we have marked the type of syntactic information/tool each method utilizes. It should be noted that these markers pertain only to the STSS algorithms themselves – a parser, for instance, cannot function without part-of-speech information, but if the STSS model using it does not explicitly employ POS tags in some other way, then only its *Parsing* column will contain a marker.

Algorithm performance was evaluated on a paraphrase detection test by using the *Microsoft Research Paraphrase Corpus* (MSRPC) [20]. This corpus consists of 5801 pairs of sentences which are all at least somewhat semantically related, but only a portion of which are true paraphrases i.e. pairs that are semantically equivalent. The entire corpus is hand-annotated with binary similarity grades determining whether a sentence pair represents a paraphrase or not. The task of STSS algorithms is to try to match the sentence-pair scores given by human annotators. Since the average inter-rater agreement among the human judges is 83%, this represents the upper boundary for the performance of an STSS system.

The main algorithm performance measure is accuracy, computed as the ratio of correctly classified sentence pairs and the total number of pairs in the corpus. Some other frequently used metrics include: precision, which represents the ratio of correctly identified paraphrases and the number of pairs classified as paraphrases by the system; recall, calculated as the ratio of correctly identified paraphrases and the total number of paraphrases in the corpus; and the F-measure which is the harmonic mean of precision and recall.

Bearing in mind that the availability of NLP tools differs from one language to another, and that in the case of minor languages many advanced tools are often non-existent, it is clear that the applicability of an STSS approach greatly depends on the syntactic information that it utilizes. This is why in Table 1 we propose a new STSS algorithm classification into three types – D, W and L – according to their applicability to languages with scarce NLP tools.

D-type algorithms are those which are *directly* applicable to all languages as they do not require any language-specific syntactic tools. *W-type* algorithms are those that have *wide* applicability to many languages since the only language-specific syntactic tool they use is a POS tagger, which is the most basic and the most widespread kind of syntax-processing tool. Finally, *L-type* algorithms are those with *limited* applicability because they rely on advanced methods of syntactic analysis, such as parsing and semantic role labeling, which are only available in a small number of languages. It should be noted that, aside from the issue of syntax usage, topological similarity approaches have to deal with the separate problem of finding a suitable knowledge base (most often in the form of a WordNet) in the desired language.

It can be seen that the number of D-type algorithms is rather low, and that they mostly utilize the statistical similarity principle. W-type algorithms are the most widespread. In contrast to the D-type methods, L-type algorithms are usually centered on a topological technique. All newer W-type approaches perform similarly to or better than D-type ones, which is to be expected since they have at their disposal more data in the form of POS tags. However, it's interesting to note that, at least within the topological similarity category, W-type methods outperform even L-type ones, which harness deeper syntactic information. One of the main reasons for this discrepancy lies in the fact that advanced syntactic tools generate significantly more errors than the simpler POS taggers, leading to inaccuracies in STSS judgments.

The best-performing D-type algorithm is the topological approach suggested by Liu et al. [15]. Among the statistical algorithms the best D-type model is Blacoe and Lapata's Simple Distributional Semantic Space [13]. The Islam and Inkpen method [10] is close to it, although the inclusion of common-word order information into its similarity score actually leads to slightly worse results. Even though the statistical approach proposed by Furlan et al. [11] is quite similar to the one in [10], the corpus used for training the COALS algorithm was rather small, which is why their results are, unsurprisingly, worse.

Lintean and Rus [12] offer the best W-type algorithm, which is also the best-performing approach in general, achieving the highest accuracy and F-measure levels. In addition, their application of the LSA metric outperforms earlier attempts of using it. It was concluded that the choice between comparing only those words with exactly the same part of speech and allowing comparisons between broader word classes depends on the task in question. Specifically, better results on the paraphrase detection test are achieved by applying the stricter restriction, thereby leveraging the additional POS information.

Topological W-type algorithms commonly perform better than the statistical ones. Although the choice of a WordNet metric to be used in a topological approach depends on various factors, two of the metrics stand out – JCN, which leads to optimal results in [14] and [19] and the second-best result in [12], and LCH, which accomplishes state-of-the-art performance levels in conjunction with [12] and the second-best result in [14]. Older approaches, like [9], attain maximal results by combining all WordNet metrics with the statistical similarity methods. However, this system requires the use of eight different measures, which makes it computationally inefficient.

Most L-type algorithms fall into the topological category, but Blacoe and Lapata [13] created a statistical distributional memory model that outperforms them. Still, even this state-of-the-art L-type method is only slightly better than the much simpler SDS approach proposed in the same paper, which makes a strong case for D-type algorithms. Furthermore, the relatively simple topological D-type model of Liu et al. [15] outmatched all L-type solutions in terms of both accuracy and the F-measure.

Wiemer-Hastings [21] showed that human similarity ratings are affected strongly by verb similarity, but less so by subject and object similarity. Most L-type algorithms tried to make use of this finding and managed to improve system accuracy by assigning different weights to similarities of different constituents. The method of Li et al. [17] achieves best results if a slightly greater weight is given to verb and noun phrases than to preposition phrases. The performance of the SyMSS model of Oliva et al. [19] increases if weighting according to semantic roles is applied, with verbs carrying the greatest weight, subjects and objects a somewhat smaller one, while adverbial complements and other roles are assigned even lower values. The topological approach of Furlan et al. [11] reaches maximal accuracy levels by giving the verbs a weight four times greater than the one used for subjects and objects. Nevertheless, this method performs poorly in comparison to the statistical algorithm presented in the same paper because many subjects and objects consist of proper nouns which cannot be found in the ConceptNet knowledge base, effectively rendering those constituents irrelevant in the calculation of the similarity score. Although the model of Lee et al. [18] is, in principle, also able to utilize subject-verb-object weighting, the effects of this technique on the system have not been explored.

VI. Conclusion

Analysis shows that the current state-of-the-art performance is achieved by using POS tags in conjunction with a topological similarity measure. The approaches which do not use any syntactic information sacrifice a degree of accuracy for simplicity and wider applicability, whereas the algorithms that utilize deep syntactic information are still hampered by the insufficient accuracies of syntax-processing tools.

This issue could be addressed by relying on simpler syntactic tools while analyzing the deeper consequences of their results. Many existing algorithms employ POS taggers either as preprocessing tools for more advanced analysis techniques, or in order to prevent words with different parts of speech from being compared. However, even though several papers have pointed out that STSS accuracy improves by taking into account the natural differences in importance between various constituents ([11], [17], [19]), no method so far has considered taking advantage of these differences on the POS level.

Assigning different weights to the similarities of different parts of speech would exploit this importance differential. Simultaneously, given the high accuracies of POS taggers, it would avoid the performance penalty generated by complex syntactic tools. Such an approach, particularly when combined with a statistical similarity algorithm, would allow for the creation of an STSS system with both high accuracy and wide applicability to languages with limited NLP resources.

References

[1] R. Barzilay and K. R. McKeown, "Sentence Fusion for Multidocument News Summarization," *Computational Linguistics*, vol. 31, no. 3, pp. 297–328, Sep. 2005.

[2] S. M. Harabagiu, S. J. Maiorano, and M. A. Paşca, "Open-Domain Textual Question Answering Techniques," *Natural Language Engineering*, vol. 9, no. 3, pp. 231–267, Sep. 2003.

[3] Z. Harris, "Distributional Structure," *Word*, vol. 10, no. 23, pp. 146–162, 1954.

[4] G. A. Miller, "WordNet: A Lexical Database for English," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, Nov. 1995.

[5] M. P. Marcus, M. A. Marcinkiewicz and B. Santorini, "Building a Large Annotated Corpus of English: The Penn Treebank," *Computational Linguistics*, vol. 19, no. 2, pp. 313–330, Jun. 1993.

[6] C. D. Manning, "Part-of-Speech Tagging from 97 % to 100 %: Is It Time for Some Linguistics?" in *Proceedings of the 12th International Conference on Computational Linguistics and Intelligent Text Processing*, 2011, pp. 171–189.

[7] E. Charniak and M. Johnson, "Coarse-to-fine n-best parsing and MaxEnt discriminative reranking," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, 2005, pp. 173–180.

[8] S. Lim, C. Lee, and D. Ra, "Dependency-based semantic role labeling using sequence labeling with a structural SVM," *Pattern Recognition Letters.*, vol. 34, no. 6, pp. 696–702, Apr. 2013.

[9] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and Knowledge-based Measures of Text Semantic Similarity," in *Proceedings of the 21st National Conference on Artificial Intelligence*, 2006, pp. 775–780.

[10] A. Islam and D. Inkpen, "Semantic Text Similarity Using Corpus-Based Word Similarity and String Similarity," *ACM Transactions on Knowledge Discovery from Data*, vol. 2, no. 2, pp. 1–25, Jul. 2008.

[11] B. Furlan, V. Sivački, D. Jovanović, and B. Nikolić, "Comparable Evaluation of Contemporary Corpus-Based and Knowledge-Based Semantic Similarity Measures of Short Texts," *Journal of Information Technology and Applications*, vol. 1, no. 1, pp. 65–72, 2011.

[12] M. Lintean and V. Rus, "Measuring Semantic Similarity in Short Texts through Greedy Pairing and Word Semantics," in *Proceedings of the Twenty-Fifth International Florida Artificial Intelligence Research Society Conference*, 2012, pp. 244–249.

[13] W. Blacoe and M. Lapata, "A Comparison of Vector-based Representations for Semantic Composition," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2012, pp. 546–556.

[14] S. Fernando and M. Stevenson, "A Semantic Similarity Approach to Paraphrase Detection," in *Proceedings of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*, 2008, pp. 45–52.

[15] X.-Y. Liu, Y.-M. Zhou, and R.-S. Zheng, "Measuring Semantic Similarity within Sentences," in *Proceedings of the 7th International Conference on Machine Learning and Cybernetics*, 2008, pp. 2558–2562.

[16] D. Ramage, A. N. Rafferty, and C. D. Manning, "Random Walks for Text Semantic Similarity," in *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing*, 2009, pp. 23–31.

[17] L. Li, Y. Zhou, B. Yuan, J. Wang, and X. Hu, "Sentence Similarity Measurement based on Shallow Parsing," in *Proceedings of the Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, 2009, pp. 487–491.

[18] M. C. Lee, J. W. Chang, T. C. Hsieh, T. I. Wang, C. Y. Su, H. H. Chen, and C. H. Chen, "A Syntactic Based Approach for Evaluating Semantics of Texts," *International Journal of Advancements in Computing Technology*, vol. 4, no. 21, pp. 220–229, Nov. 2012.

[19] J. Oliva, J. I. Serrano, M. D. del Castillo, and Á. Iglesias, "SyMSS: A syntax-based measure for short-text semantic similarity," *Data & Knowledge Engineering*, vol. 70, no. 4, pp. 390–405, Apr. 2011.

[20] B. Dolan, C. Quirk, and C. Brockett, "Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources," in *Proceedings of the 20th International Conference on Computational Linguistics*, 2004, Article No. 350.

[21] P. Wiemer-Hastings, "All parts are not created equal: SIAM-LSA," in *Proceedings of 26th Annual Conference of the Cognitive Science Society*, 2004.