Effects of Graph Generation for Unsupervised Non-Contextual Single Document Keyword Extraction

Natalie Schluter
Center for Language Technology, University of Copenhagen, Copenhagen, Denmark
natschluter@hum.ku.dk

Abstract. This paper presents an exhaustive study on the generation of graph input to unsupervised graph-based non-contextual single document keyword extraction systems. A concrete hypothesis on concept coordination for documents that are scientific articles is put forward, consistent with two separate graph models : one which is based on word adjacency in the linear text–an approach forming the foundation of all previous graph-based keyword extraction methods, and a novel one that is based on word adjacency modulo their modifiers. In doing so, we achieve a best reported NDCG score to date of 0.431 for any system on the same data. In terms of a *best parameter* f-score, we achieve the highest reported to date (0.714) at a reasonable ranked list cut-off of $n = 6$, which is also the best reported f-score for any keyword extraction or generation system in the literature on the same data. The best-parameter f-score corresponds to a reduction in error of 12.6% conservatively.

1 Introduction

Recent and state-of-the-art approaches to unsupervised non-contextual single document keyword extraction typically work on some sort of graph of the input text, formed with respect to word order. The graphs generally pose word-forms as nodes and place edges between words so long as their proximity in the linear text is within some threshold d ; however the characteristics of the edges, and the deletion or association of certain nodes (in both pre- and post-processing) vary from one approach to the other and no exhaustive study of these choices has been motivated or exhaustively tested using a single *relevant* measure. Similarly, the use of the linear order of words in the text as the basis for the edge relations in text graphs has been loosely associated with a linguistic syntax motivation, however no more sophisticated accounting of syntactic relations has been attempted to date. The work presented in this paper partially bridges this gap.

We present an exhaustive study on the generation of graph input to unsupervised graph-based non-contextual single document keyword extraction systems. We consider the question of graph motivation, and put forward a concrete hypothesis on concept coordination for documents that are scientific articles. Corresponding to the requirements of the graph model, we consider two types of relations between words for such a graph, one which is based on word adjacency in the linear text–an approach forming the foundation of all previous graph-based keyword extraction methods, and a novel one that is based on word adjacency modulo their modifiers. In doing so, we achieve a best reported NDCG score to date of 0.431 for any system on the same data. In terms of a *best parameter* f-score, we achieve the highest reported to date (0.714) at a reasonable ranked list cut-off of $n = 6$, which is also the best reported f-score for any keyword extraction or generation system in the literature on the same data. The f-score corresponds to a reduction in error of 12.6%, or even more if we set both systems to a ranked list cut-off of $n = 6$ (since the previous best f-score was achieved at a best parameter of $n = 9$). (This latter score is also reproduced in Table 1.)

Following some preliminaries on the definition of the task, we discuss previous work in unsupervised non-contextual single document keyword extraction (Section 2). We present the graph models we investigate, in Section 3, which is the main contribution of this paper. Section 4 reviews the centrality measures used. Finally, we present the evaluation of the resulting systems (Section 5), followed by a brief discussion of conclusions and open problems (Section 6).

2 Preliminaries

We identify two broad types of *single document keyword extraction* (SDKE). *Contextual SDKE* makes use of the document set to which the relevant document belongs, and in which there are similar documents ; other information outside of the document set may also be used in some types of contextual SDKE. *Non-contextual SDKE* makes use of only the relevant document with no other information. The latter does not necessarily make the assumption of independence of documents in general. In fact, non-contextual SDKE is important for the case of isolated documents (not part of a document set), as well as for documents for which relevant supplementary information may be non-existent or unreliable.

Undirected Graphs A simple *graph* G is a pair (V, E), where V is the set of vertices and $E \subseteq V \times V$ is the set of edges (where the pairs of vertices are unordered). The edge uv is said to be *incident* with the vertices u and v. A *multi-graph* is a graph where there may be more than one edge between two vertices (the edge set is a multi-set). The *degree* $deg(v)$ of a vertex v is then the number of distinct edges with which it is incident. A *walk* of length k from vertex u to vertex v is a sequence of k edges, $v_1v_2, v_2v_3, \ldots, v_{k-1}v_k, v_kv_{k+1}$ and a *path* from u to v is a walk from u to v where no edge is repeated. Finally, a *complete graph* is a graph with all $|V|(|V| - 1)/2$ possible edges, and a *clique* is a subgraph that is complete.

2.1 Previous Work

Published work including some discussion of text graph representations for non-contextual SDKE has considered only (1) the directed-/undirected-ness of edges on stop-word filtered graphs and (2) different proximity thresholds d for placing these edges between word nodes (Mihalcea & Tarau, 2004; Litvak & Last, 2008; Litvak *et al.*, 2013; Rose *et al.*, 2010; Schluter, 2014; Boudin, 2013). [1] The proximity threshold of $d = 1$ was found by (Mihalcea & Tarau, 2004) to perform best, and all other research on the task has consistently maintained this threshold ; the present work follows suit in that respect.

In terms of the unsupervised non-contextual single document keyword extraction task itself, (Mihalcea & Tarau, 2004) had the pioneering work, also introducing graph-based techniques for this task with the application of PageRank (Page *et al.*, 1999). (Litvak & Last, 2008) follow this approach, but apply HITS instead (on a different dataset). Finally, (Rose *et al.*, 2010) observed that using the simple degree of a vertex in the network produced what were at the time state-of-the-art results (with precision 0.337, recall 0.415 and f-score 0.372, at a ranked list cut-off of $n = \frac{1}{3}N$, where N is the number of words in the document, on the Inspec corpus) ; the technique was later re-discovered by (Litvak *et al.*, 2013) and (Boudin, 2013).

3 Graph generation and other pre-processing

In this section we present the graph model of the document text as well as two consistent instances of this model : adjacency graphs and parse graphs.

3.1 Graph Model

We follow the document model proposed in (Schluter, 2014) for keyword extraction, which proposes a graph model from the point of view of document *synthesis* (as opposed to the document *analysis* model proposed by (Mihalcea & Tarau, 2004)). In generating scientific text on a given topic (or given related topics), the "author" may require other concepts to regularly support the discussion (for example, definitions or explanations) ; this is a sort of concept coordination. Two basic assumptions are adopted about this concept coordination in the model. The first assumption is that the author is communicating in the most efficient manner possible, and that supporting concepts are named only when absolutely necessary. The second assumption is that in supporting or defining a concept, textual mention of a topic concept and supporting concepts should occur rather "close" to each other, in terms of the linear order of concepts (words) in the texts. These concept support relations are approximated therefore by co-occurrence relations–relations that are essentially symmetric (undirected) : there is no clear order that should be observed between topic concepts and supporting concepts within a single sentence (or over several sentences for that matter). We note that the network is *not* the meaning of the documentation ; rather it is a *representation of its construction*. Flow through the concept network is seen as *communicative*–concept-building on the part of the author for the reader.

[1] (Litvak & Last, 2008) motivate their choice of directed graphs by the extensive clustering and classification results-driven graph study presented in (Schenker *et al.*, 2005), but do not motivate the choice with respect to the keyword extraction task they undertake.

As such, we model text as an *undirected* graph, where vertices are words appearing in the text and edges model the concept coordination relationships discussed above. There are many methods of producing (undirected) edges for our graph that are compatible with the model described above. We consider two plausible ones for this paper. In Section 3.3 we describe a graph model similar to that of (Mihalcea & Tarau, 2004; Litvak & Last, 2008; Rose *et al.*, 2010; Litvak *et al.*, 2013) and in Section 3.4 we propose a novel graph model created out of parse graphs of document sentences. First we discuss the pre-processing of the text carried out prior to graph construction, as well as graph parameters that common to both types of graphs.

3.2 Pre-processing and common graph parameters

Preprocessing. For both main types of models, we first carry out sentence detection, tokenisation and part-of-speech tagging on the corpus, using the Stanford POS Tagger (Toutanova *et al.*, 2003). We remove all punctuation from individual sentences.

Filtering out stop-words. For both main models, we construct reduced and full graphs. Their exact manner of construction is specific to the graph type (adjacency or parse) (Cf. Sections 3.3 and 3.4).

1. The **reduced** graph contains the text stripped of stop-words, in order to have edges reflect relationships between semantically full words more directly, resulting in a denser graph. For the remaining non-stop-words, words of the same form and part-of-speech are merged into a single node.
2. The **full** graph is constructed from the full text. However, it contains a special type of node for stop-words. Nodes decorated with distinct non-stop-words of the same form and part-of-speech are still merged into a single node, but nodes decorated with stop-words are never merged, resulting in a sparser graph. The centrality measure, rather than the pre-processing is left with the full burden of ranking the important words. Stop-words are generally words that occur frequently in text, so by not merging identical ones into single nodes, we hope to prevent the centrality measure of choice from finding these units important.

Edge multiplicity. For both adjacency graphs and parse graphs we test their **simple** and **multi**-graph versions. Edges from both graphs reflect of course relations between words, but the multi-graph versions are meant to also reflect frequency of concept coordination.

Note however that for some centrality measures, there is no difference between multi- and simple graphs (Cf. Section 4).

3.3 Document adjacency graphs

Document adjacency graphs model text linear relationships between words (i.e., that they are beside or close to each other in the text). As such, for full (reduced) document adjacency graphs, an edge between two words is added to the graph if these two words are adjacent in the (stop-word filtered) text. There are therefore four different document adjacency graph models that we investigate, considering all common graph parameters combinations.

The generated reduced and full adjacency text graphs for Ex 1 below are given in the top of Figure 1.

(Ex 1) Compatibility of systems of linear constraints over the set of natural numbers. Criteria of compatibility of a system of linear Diophantine equations, strict inequations, and nonstrict inequations are considered. Upper bounds for components of a minimal set of solutions and algorithms of construction of minimal generating sets of solutions for all types of systems are given. These criteria and the corresponding algorithms for constructing a minimal supporting set of solutions can be used in solving all the considered types systems and systems of mixed types. [2]

3.4 Document parse graphs

It is straightforward that a better approximation of concepts can be achieved by first organising sub-strings of a sentence into units observing the communicative flow between units and sub-units. For English, the dependency tree syntactic re-

[2] This is abstract 1939 from the test files in the Inspec corpus, first used as an example in (Mihalcea & Tarau, 2004).

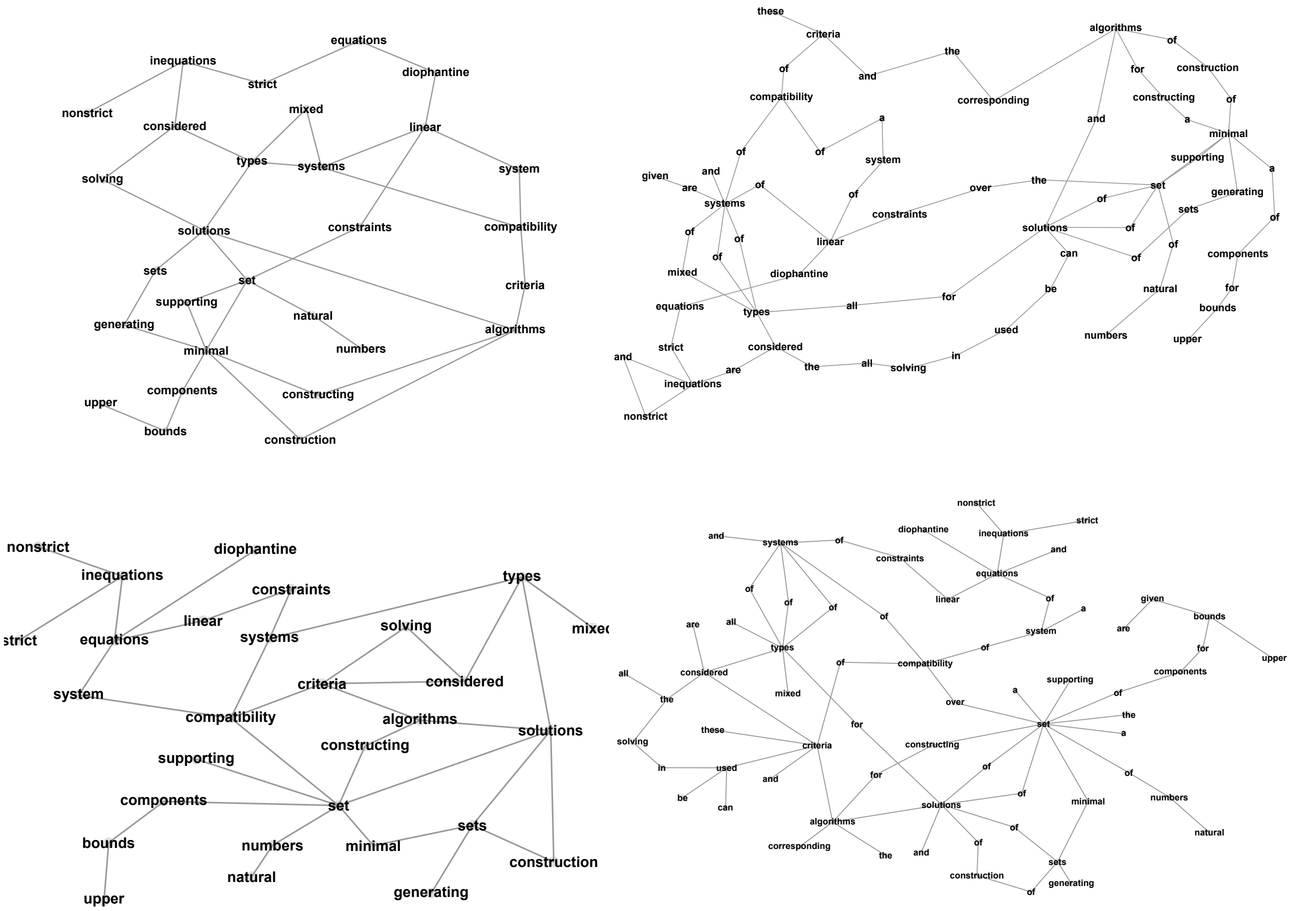

FIGURE 1 – Simple adjacency graphs (top), basic parse graphs (bottom) for Ex 1. Reduced graphs are on the left and the full graphs on the right.

presentation of sentences provides exactly the organisation of a sentence that we need to model this. The set of dependency trees of a document are organised into what we refer to as the *document parse graph*.

Following preprocessing (and before any stop-word filtering), the text is sent through the Stanford parser to obtain the associated dependency parses (basic dependency conversion) (de Marneffe *et al.*, 2006). This yields two additional graph models that we investigate (considering all other common graph parameter combinations).

The construction of full parse graphs is similar to that of adjacency graphs. The construction of reduced parse graphs is slightly more involved. By contracting stop-word vertices some information can potentially be lost : the children of stop-words become closer to their original grandparent than they are to each other, which is not the intended model. In an attempt at circumventing this effect, we first create a clique out of the children of stop-words before contraction.

For Ex 1, the generated reduced and full parse are given at the bottom of Figure 1.

3.5 Post-processing.

We carry out similar post-processing to (Mihalcea & Tarau, 2004). That is, sequences of adjacent keywords from the text are possibly collapsed into a multi-word keyword, depending on their scores. We score a multi-unit keyword by the maximum score of words they are composed with ; we also tried the using the average score of word components, but with worst performance and so do not report these scores here. This yields a candidate list where there may be unit overlaps in keywords. We therefore test an extra post-processing step which keeps only the keyword with the highest score among two overlapping keywords (this corresponds to **excl** (as opposed to **incl**) in results Tables 1-3). Ties are broken with a preference for longer keywords ; moreover, the proposed keywords must not start or end with a stop-word, and must be grounded in a noun (i.e., the rightmost word of a multi-word keyword must be a noun). Keywords consisting of at most three words are considered.

4 Centrality measures

(Schluter, 2014) investigates seven different centrality measures for ranking nodes in undirected graphs, two of which corresponded to previously published state-of-the-art approaches to non-contextual SDKE among the centrality measure classes of degree-like centrality, closeness centrality and betweenness centrality, outlined by (Borgatti & Everett, 2006). This work showed that closeness centrality measures, which rank nodes according to their general (minimum) distance to other nodes in the graph, did not perform as well as degree-like centrality measures or betweenness centrality measures. In this paper we consider the performance of systems based on different graphs across the three betweenness centrality measures as well as the degree-like centralities.

The *degree centrality* of a vertex v in a graph G is simply its degree $deg(v)$. Within the context of text graphs, this is a measure of how much of a first-hand support a text vertex (concept) is for other text vertices (concepts).

The *eigenvector centrality* is essentially the deterministic version of the PageRank algorithm (Page *et al.*, 1999) for *undirected* graphs, as well as the output of the HITS algorithm (for *directed* graphs) upon convergence (provided all eigenvalues are distinct) (Kleinberg, 1999). The eigenvector centrality of a node $v_i \in V(G)$, $C_{EI}(v_i)$ is found by calculating the principal eigenvector of the adjacency matrix for the graph. The ith entry in this vector is $C_{EI}(v_i)$. To bypass connectivity issues, we use the PageRank "teleportation trick", transforming the input graph into a complete graph, simply incrementing the weight of all possible edges by 1.

The *betweenness centrality* of a vertex quantifies how often a node acts as a bridge along the shortest path between two other nodes. In the context of our text graph, the betweenness centrality can be seen as a measure of how the presentation of a scientific subject must employ a given word (concept) as support when moving the discussion between two different concepts. We consider three different betweenness centrality measures.

The *(normalised) betweenness centrality* $C_B(x)$ for vertex x is defined as $C_B(x) := \sum_{s \in V(G)} \sum_{t \in V(G)} \frac{\sigma_{st}(x)}{\sigma_{st}}$, where σ_{st} is the number of shortest paths between nodes s and t.[3]

$C_B(x)$ gives more weight to pairs of vertices at a larger distance from each other. If one wishes to consider all shortest paths to contribute the same weight, one approach is to normalise by the shortest distance between s and t, which yields *length-scaled betweenness centrality*, $C_{LSB}(x) : C_{LSB}(x) := \sum_{s \in V(G)} \sum_{t \in V(G)} \frac{\sigma_{st}(x)}{d(s,t)\sigma_{st}}$.

Finally, the *distance-weighted fragmentation* $C_{DWF}(x)$ of vertex x measures the fragmentation of a graph if we took x out of it. It is defined as $C_{DWF}(x) := C_{DWF}(G - x) - C_{DWF}(G)$, where $C_{DWF}(G) := 1 - \frac{2\sum_{i \neq j} \frac{1}{d(i,j)}}{n(n-1)}$. Note that $G - x$ (the graph obtained from G by removing vertex x and any edges incident to x) should be more fragmented than G. (We also shift all scores, so that they are positive.)

Note that by these definitions, there will be no difference in betweenness centrality measure results on simple graphs versus multi-graphs.

5 Experiments and Evaluation

We carry out our experiments on the test set from the Inspec abstract corpus (Hulth, 2003) consisting of 500 abstracts for scientific articles, along with the *uncontrolled* corresponding keywords.

We evaluate the systems across the variety of graph inputs in terms of average standard Normalised Discounted Cumulative Gain (NDCG). We also provide best parameter (for ranked list cut-off n) precision, recall and f-score, which informs us of when a system reaches its optimality, rather than providing any general system evaluation. Document keyword sets are relatively small, but not too small, so if n is too small or too large when it reaches a good optimal f-score, the system cannot necessarily be considered successful. On the other hand, if for example $n \in \{5, \ldots, 10\}$ and it reaches a global optimum among all systems, it is easier to argue this to be a success.

The results are reported in Tables 1 through 3. We observe that best scores according to both metrics are generally achieved by parse graphs, suggesting that the parse graph model is superior to the simple adjacency graph model.

[3] In fact, the normalised version of betweenness centrality normalises $C_B(x)$ by the number of pairs of nodes in the graph. However, since we are not comparing two different graphs, but only two different nodes of the same graph, this expression of betweenness centrality has the same power as its normalised version.

We observe that degree centrality has the best NDCG score of 0.431, for the parse full graph. We note that these NDCG scores differ from those of (Schluter, 2014) which were erroneous due to a bug in the evaluation software. The best f-score of 0.714 among all models is achieved by the parse graph under the distance-weighted fragmentation measure, at a cut-off of $n = 6$, which is very reasonable for this task ; however, curiously this model (and measure) achieves a relatively poor NDCG score, which indicates that after this ideal cut-off, the ranking system fails.

With these degree-centrality results, we can observe differences between simple and multi-graphs, and more so for reduced graphs than for full graphs. This makes sense since we never merge stop-word nodes in full graphs and thereby account for a type of co-occurrence frequency via stop-words. Still the differences in simple and multi-graph scores are relatively small, perhaps contrary to intuition. We hypothesis this to be the result of the nature of the document set in question ; the documents are very short and therefore contain less repeat co-occurrences.

graph	pre-p	post-p	n	prec	rec	f1	NDCG
adj	reduced	incl	14	0.357	0.625	0.455	0.418
		excl	13	0.308	0.5	0.381	0.390
	full	incl	18	0.333	0.75	0.462	0.410
		excl	12	0.333	0.5	0.4	0.383
parse	reduced/	incl	6	0.833	0.625	**0.714**	0.406
	full	excl	5	0.6	0.375	0.462	0.374

graph	pre-p	post-p	n	prec	rec	f1	NDCG
adj	reduced	incl	8	0.5	0.5	0.5	0.399
		excl	2	1.0	0.25	0.4	0.386
	full	incl	8	0.5	0.5	0.5	0.398
		excl	2	1.0	0.25	0.4	0.385
parse	reduced	incl	8	0.5	0.5	0.5	0.414
		excl	2	1.0	0.25	0.4	0.403
	full	incl	8	0.5	0.5	0.5	0.412
		excl	2	1.0	0.25	0.4	0.401

TABLE 1 – Distance-weighted fragmentation results (left). Betweenness centrality results (right). The scores for multigraph are precisely the same.

graph	pre-p	post-p	n	prec	rec	f-score	NDCG
adj	reduced	incl	14	0.429	0.75	0.545	0.397
		excl	11	0.364	0.5	0.421	0.378
	full	incl	11	0.455	0.625	0.526	0.395
		excl	2	1.0	0.25	0.4	0.375
parse	reduced	incl	12	0.5	0.75	**0.6**	0.417
		excl	10	0.4	0.5	0.444	0.406
	full	incl	11	0.455	0.625	0.526	0.417
		excl	11	0.364	0.5	0.421	0.408

graph	pre-p	post-p	n	prec	rec	f1	NDCG
adj	reduced	incl	20	0.3	0.75	0.429	0.419
		excl	15	0.267	0.5	0.348	0.395
	full	incl	19	0.316	0.75	0.444	0.413
		excl	14	0.286	0.5	0.363	0.388
parse	reduced	incl	13	0.385	0.625	0.476	0.426
		excl	17	0.235	0.5	0.32	0.376
	full	incl	17	0.353	0.75	0.48	0.419
		excl	14	0.286	0.5	0.364	0.367

TABLE 2 – Length scaled betweenness centrality results (left). The scores for simple and multi-graphs are precisely the same. Eigenvector centrality results (right).

edge	pre-p	post-p	n	prec	rec	f1	NDCG
mult							
simple	reduced	incl	5	0.6	0.375	0.462	0.423
		excl	14	0.286	0.5	0.363	0.401
	full	incl	5	0.6	0.375	0.462	0.415
		excl	14	0.286	0.5	0.363	0.392
multi	reduced	incl	19	0.316	0.75	0.444	0.423
		excl	14	0.286	0.5	0.363	0.402
	full	incl	19	0.316	0.75	0.444	0.418
		excl	14	0.286	0.5	0.363	0.396

edge	pre-p	post-p	n	prec	rec	f1	NDCG
mult							
simple	reduced	incl	11	0.455	0.625	0.526	**0.431**
		excl	9	0.333	0.375	0.353	0.381
	full	incl	5	0.6	0.375	0.462	0.421
		excl	16	0.25	0.5	0.333	0.370
multi	reduced	incl	11	0.455	0.625	0.526	0.427
		excl	9	0.333	0.375	0.353	0.382
	full	incl	5	0.6	0.375	0.462	0.419
		excl	16	0.25	0.5	0.333	0.372

TABLE 3 – Degree centrality results for adjacency graphs (left) and parse graphs (right).

6 Conclusions and open questions

We have introduced a novel parse text graph for the representation of documents that is shown to perform better in non-contextual single document keyword extraction, producing the highest reported NDCG score to date, as well as the highest best parameter f-score. In our opinion this model is more language independent than the adjacency graph document model, as it relies slightly less on sentential linear order ; this is an open question for future investigation. In addition, the question as to whether the multi-graph version of document graph models helps systems when the input are larger documents remains open ; for smaller documents the answer seems to be negative.

Références

BORGATTI S. P. & EVERETT M. G. (2006). A graph-theoretic perspective on centrality. *Social Networks*, **28**(4), 466 – 484.

BOUDIN F. (2013). A comparison of centrality measures for graph-based keyphrase extraction. In *Proc. of IJCNLP 2013*, Nagoya, Japan.

DE MARNEFFE M.-C., MACCARTNEY B. & MANNING C. D. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, p. 449–454.

HULTH A. (2003). Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, p. 216–223, Stroudsburg, PA, USA : Association for Computational Linguistics.

KLEINBERG J. M. (1999). Authoritative sources in a hyperlinked environment. *J. ACM*, **46**(5), 604–632.

LITVAK M. & LAST M. (2008). Graph-based keyword extraction for single-document summarization. In *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, MMIES '08, p. 17–24, Stroudsburg, PA, USA : Association for Computational Linguistics.

LITVAK M., LAST M. & KANDEL A. (2013). Degext : a language-independent keyphrase extractor. *J. Ambient Intelligence and Humanized Computing*, **4**(3), 377–387.

MIHALCEA R. & TARAU P. (2004). Textrank : Bringing order into text. In *Proceedings of EMNLP*, p. 404–411.

PAGE L., BRIN S., MOTWANI R. & WINOGRAD T. (1999). *The PageRank Citation Ranking : Bringing Order to the Web.* Technical Report 1999-66, Stanford InfoLab. Previous number = SIDL-WP-1999-0120.

ROSE S., ENGEL D., CRAMER N. & COWLEY W. (2010). *Automatic Keyword Extraction from Individual Documents*, In *Text Mining. Applications and Theory*, p. 1–20. John Wiley and Sons, Ltd.

SCHENKER A., LAST H. & KANDEL M. (2005). *Graph-Theoretic Techniques for Web Content Mining*, volume 62 of *Series in Machine Perception and Artificial Intelligence*. World Scientific.

SCHLUTER N. (2014). Centrality measures for non-contextual graph-based unsupervised single document keyword extraction. In *Proc. of TALN 2014*.

TOUTANOVA K., KLEIN D., MANNING C. D. & SINGER Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, Stroudsburg, PA, USA : Association for Computational Linguistics.