A New Lexical Chain Algorithm Used for Automatic Summarization

Edgar GONZÀLEZ [1], Maria FUENTES
TALP Research Center, Spain

Abstract. Lexical chains are widely used as a representation of text for Natural Language Processing tasks. However, efficient algorithms for the construction of lexical chains often resort to local decisions. We propose a new algorithm for Lexical Chaining, based on a global function optimization through Relaxation Labelling. A preliminary evaluation of the performance of our approach has been performed on a Catalan agency news corpus. The comparison to an efficient state-of-the-art algorithm for Lexical Chaining gives promising results. The resulting lexical chainer has been used for a complete multilingual Automatic Summarization system, available on-line.

Keywords. Lexical Chains, Automatic Summarization, Relaxation Labelling, Multilingual Technologies

Introduction

Lexical chains try to identify cohesion links between parts of text by identifying relations holding between their words. Two pieces of text are considered to be lexically related not only if they use the same words, but also if they use semantically related words. This is a way to obtain certain structure of a text based on the distribution of its content.

A sample text fragment annotated with lexical chains can be seen in Figure 1. In this example, the words *representant* (*representative*) and *candidat* (*candidate*) are linked because their senses share a common hypernym, *person*; whereas *discussió* (*discussion*) is a direct hypernym of *debat* (*debate*). Multiple occurrences of a single word like *candidat* are linked by reiteration. Words in a chain can also be linked by other relations, such as synonymy, antonymy, holonymy or meronymy.

Lexical chains provide a representation of text that has been widely used in a variety of Information Retrieval and Natural Language Processing tasks, from text segmentation [1,2] to Word Sense Disambiguation [3,4,5], term weighting for IR tasks [6], topic detection [7], detection of malapropisms [8], hypertext generation [9], detecting topic and sub-topic shifts in texts [3,6,10,11], analysis of the structure of texts to compute their similarity [12], Passage Retrieval [13], Question & Answering [14] and Automatic Summarization [4,15,16].

At the core of all these applications lies a lexical chain construction algorithm. A detailed overview of approaches to lexical chain identification and use can be found

[1]Corresponding Author: Edgar Gonzàlez, Despatx S107, Edifici Omega, Campus Nord, c/Jordi Girona, 1-3, 08034 Barcelona, Spain; E-mail: egonzalez@lsi.upc.edu.

En una trobada pública en la casa de campanya de Cárdenas i enfront dels **representants**$_1$ dels mitjans, els tres **candidats**$_1$ van discutir durant unes dues hores les seues propostes sobre el **debat**$_2$. El **candidat**$_1$ del PAN va insistir reiteradament a celebrar esta mateixa nit esta **discussió**$_2$...

Figure 1. Sample Text Fragment Annotated with Lexical Chains

in [11]. Many among the proposed algorithms are greedy in nature, and are hence unable to recover from wrong decisions in early stages. In [4], a non-greedy strategy is proposed, but the approach has the drawback of an exponential runtime cost. The approaches of [17] and [5] give linear cost approximations of the behavior of [4]. However, although their results are better than those of greedy algorithms, they lack a global optimization criterion, and they end up resorting to local decisions, which may lead to non-optimal solutions.

In this context, global function optimization algorithms can be of great help. Among them, Relaxation Labelling techniques are a popular choice, and have been used for other Natural Language Processing tasks, such as POS tagging [18].

In this paper we present a new algorithm, RELAX, for the construction of lexical chains, based on Relaxation Labelling. This algorithm lies at the core of a multilingual Summarization system instantiating FEMSUM, a Flexible Multitask Summarization architecture implemented using flexible, reusable and open source Natural Language Processing components. An online demo of the built system has also been made available.

The organization of the rest of the paper is as follows. Section 1 describes the FEMSUM architecture and how our system behaves as one of its instantiations. Section 2 formalizes the problem of Lexical Chaining and gives its reduction to Relaxation Labelling, and Section 3 describes experiments carried to test the validity of our approach. Section 4 gives implementation details of our system. Section 5 draws conclusions of our work.

1. FEMSUM Architecture Instantiation

FEMSUM is a flexible and highly modularized Multitask Summarization architecture, on which parametrizable Summarizers suitable to specific needs can be built [19]. The architecture is divided in four main components:

1. Linguistic Processor
2. Relevant Information Detector
3. Content Extractor
4. Summary Composer

In our instantiation of FEMSUM, the FreeLing suite (see Section 4) is used as Linguistic Processor, and Lexical Chaining works as Relevant Information Detector.

For Lexical Chaining, we follow an approach similar to [4]. After linguistic processing, lexical chain candidates can be selected among common nouns, proper nouns, named entities, verbs, adjectives and/or adverbs. No previous Word Sense Disambiguation is performed. For each chain candidate, three kinds of relations are considered:

- *Extra-Strong*, between a word and its repetition.

- *Strong*, between two words connected by a direct semantic relation.
- *Medium-Strong*, between two words connected by a path of semantic relations.

We have used the WordNet or EuroWordNet lexical databases, depending on the language, as source for semantic relations.

Once chains are identified, in the Content Extractor component they are scored according to a number of heuristics: their length, the kind of relation between their words, their starting position within the text... After scoring, the average of the scores of all chains μ_s and the corresponding standard deviation σ_s are found, a threshold value $\tau = \mu_s + 2 \cdot \sigma_s$ is determined, and the chains c_k are classified into:

- *Strong*, if their score $s(c_k)$ is greater or equal than τ.
- *Medium*, if $s(c_k)$ lies between τ and $\tau/2$.
- *Light*, if $s(c_k)$ is less than $\tau/2$.

These values are heuristic, but were found to work well in the considered corpora [19]. It is also important to note that the classification of chains into *Strong*, *Medium* and *Light* is not directly related to that of their contained relations into *Extra-Strong*, *Strong* and *Medium-Strong*.

Finally, the last component, the Summary Composer, uses a set of heuristics to incorporate sentences crossed by *Strong* chains, generating an extractive summary until it exceeds the demanded size. However, in contrast to other approaches where lexical chains are used, *Medium* and *Light* chains may be considered in addition to the typical *Strong* ones. That can be specially useful in domains like spontaneous speech, were the distribution of the frequency of words is rather skewed and *Strong* chains tend to provide a misrepresentation of the information in a text.

As seen, the proposed Summarizer has a number of parameters than can be modified. The chain candidate words (common nouns, proper nouns, named entities, verbs, adjectives and/or adverbs), the kind of semantic relations used for Lexical Chaining (from *Extra-Strong* only to all the considered types), and the kind of chains used by the Summary Composer (from *Strong* only to all of them), as well as the chain scoring and heuristics used, can be adjusted to tune the system to specific media, domain and language.

More details of the overall summarization process can be found in [19].

2. Lexical Chaining

As mentioned, our setting for Lexical Chaining follows that of [4]. However, we only consider the semantic relations of synonymy and hypernymy/hyponymy, given that other kinds of relations have not been representatively annotated within EuroWordNet. The inclusion of antonymy and holonymy/meronymy relations, used in [4], should nevertheless be straightforward.

Our setting also uses the *one sense per discourse* assumption for polysemous words. Given that this assumption will only hold for short documents, the use of a tiling algorithm such as [10] is necessary for longer documents, to allow for lexical chains to be resolved locally within each text segment, and then merged across them.

Next Section 2.1 formalizes the considered problem of Lexical Chaining, and Section 2.2 gives an overview of the Relaxation Labelling algorithm. Taking both concepts into account, a reduction of Lexical Chaining to Relaxation Labelling is given in Section 2.3.

2.1. Problem Formalization

Within our setting, the problem of Lexical Chaining can be formalized as an optimization problem. Given a set of words $W = \{w_i\}$ and a set of senses $S = \{\sigma_j\}$, we can define a *meaning* function ϕ mapping words to sets of senses $\phi(w_i) = \{\sigma_{i1} \dots \sigma_{im_i}\} \subset S$. We can assume that $\forall w_i : \phi(w_i) \neq \emptyset$.

Additionally, we consider an irreflexive and asymmetric *direct hypernymy* relation defined on the set S, and we will note $\sigma_1 \rightarrow \sigma_2$ iff σ_1 is a direct hypernym of σ_2. The *hypernymy* relation is the transitive closure of the *direct hypernymy* relation, and we will note $\sigma_1 \stackrel{\star}{\rightarrow} \sigma_2$ iff σ_1 is a hypernym of σ_2. This *hypernymy* relation must be a partial ordering on the set S. Additionally, we can define a *strict hypernymy* relation, noted by $\sigma_1 \stackrel{+}{\rightarrow} \sigma_2$, and equivalent to $\sigma_1 \stackrel{*}{\rightarrow} \sigma_2 \wedge \sigma_1 \neq \sigma_2$. In our experiments, the set of senses will correspond to synsets in WordNet/EuroWordNet, and the hypernymy relation there defined will be used.

For the purposes of lexical chain construction, a document (or segment) can be seen as a sequence of words $D = (w_1 \dots w_n)$, with $w_i \in W$. Words that are not considered for lexical chain computation can be omitted from this representation. An *interpretation* function ψ then maps each word w_i in D to one of its senses $\psi(w_i) = \sigma_j \in \phi(w_i)$. Last, a lexical chaining C is a partition of document D into a set $\{c_1 \dots c_k\}$ of disjoint chains, $c_m \subset D$.

We can then define a score function $s(\sigma_j, \sigma_{j'})$ for a pair of senses, whose value can be t_{syn}, t_{hyp} or 0, according to whether the two senses are the same, one is an hypernym of the other, or they have no relation. t_{syn} and t_{hyp} are hence parameters set to specify the weight of synonymy and hypernymy in lexical chains. The score of a chain c_m under an interpretation ψ is then the sum of the scores of the senses assigned by ψ to each pair of words in the chain (irrespectively of order of words within the pair):

$$s(c_m \mid \psi) = \sum_{w_i, w_{i'} \in c_m} s(\psi(w_i), \psi(w_{i'}))$$

The score $s(C \mid \psi)$ of a lexical chaining C under interpretation ψ is the sum of the scores of all its chains.

The aim of Lexical Chaining is to find the *optimal* interpretation $\widehat{\psi}$ and the *optimal* lexical chaining $\widehat{C}$ giving the maximum score $s(\widehat{C} \mid \widehat{\psi})$. Additionally, the chaining $\widehat{C}$ with maximum cardinality is preferred, to ensure that non-related words are kept in different chains.

2.2. Relaxation Labelling

Relaxation Labelling is a generic name for a family of iterative algorithms which perform function optimization, based on local information [20].

A *labelling problem* is a tuple (V, L, R), consisting of a set of *variables* $V = \{v_1 \dots v_n\}$; a set, $L = \{L_1 \dots L_n\}$, of sets of possible *labels* for each v_i, $L_i = \{l_{i1} \dots l_{im_i}\}$; and a set of *constraints* $R = \{r_1 \dots r_r\}$, where each *constraint* r_k is a sequence of variable-label pairs (v_{ki}, l_{kij}) and an associated compatibility value t_k: $r_k = (t_k, (v_{k1}, l_{k1j_1}) \dots (v_{kz_k}, l_{kz_kj_{z_k}}))$.

We can define a *weighted labelling* P as an assignation of a weight $p(v_i, l_{ij})$ to each label l_{ij} of each variable v_i, with $\forall i : \sum_{j=1}^{m_i} p(v_i, l_{ij}) = 1$. The aim of Relaxation La-

belling is then to find a weighted labelling $\widehat{P}$ such that global consistency is maximized. This means maximizing, for each variable v_i, the average *support* for that variable $S(v_i)$, defined as the weighted sum of the *support* $S(v_i, l_{ij})$ received by each of its labels:

$$S(v_i) = \sum_{j=1}^{m_i} p(v_i, l_{ij}) \cdot S(v_i, l_{ij})$$

It is important to note that Relaxation Labelling is a vector optimization algorithm, and it is the support of each variable which is optimized, not a combination (such as a sum or product) of all of them.

Several support functions $S(v_i, l_{ij})$ can be considered. For our purposes, it will be convenient to consider the sum of the influences of the constraints r_k which include the pair (v_i, l_{ij}), where the influence of a constraint r_k for the label l_{ij}, $Inf(r_k, v_i, l_{ij})$, is the product of the constraint compatibility times the product of weights of each one of the variable labels involved in the constraint, excluding the one under consideration:

$$S(v_i, l_{ij}) = \sum_{r_k : (v_i, l_{ij}) \in r_k} Inf(r_k, v_i, l_{ij})$$

$$Inf(r_k, v_i, l_{ij}) = t_k \cdot p(v_{k1}, l_{k1j_1}) \cdot \ldots \cdot p(v_{kz_k}, l_{kz_kj_{z_k}})$$

where the term $p(v_i, l_{ij})$ is excluded in the $Inf(r_k, v_i, l_{ij})$.

Relaxation Labelling finds the optimal weighted labelling by iterative refinement from a starting solution P_0. The advantages of the algorithm include its expressiveness, flexibility and robustness, while the drawbacks are its potential computational cost, the local nature of the found maxima, and the lack of a warranty of convergence.

2.3. Problem Reduction

To reduce the Lexical Chaining problem proposed in 2.1 to a labelling problem, the following steps are applied:

- Each word in the document w_i is considered a variable, and each of its possible senses σ_j according to the meaning function $\phi(w_i)$ is considered a label for it.
- For each pair of words $w_i, w_{i'}$:
  - If there is a sense σ_j that is shared between the meanings of the two words, a constraint is created with the form: $r_k = (t_{syn}, (w_i, \sigma_j), (w_{i'}, \sigma_j))$.
  - If there is a sense σ_j of w_i that is a strict hypernym of a sense $\sigma_{j'}$ of $w_{i'}$, or vice-versa, a constraint is created with the form: $r_k = (t_{hyp}, (w_i, \sigma_j), (w_{i'}, \sigma_{j'}))$.

It can be shown that the sum of supports of all considered variables, as defined in Section 2.2, is closely related to the score function of Lexical Chaining, as defined in Section 2.1. Hence, the optimal weighted labelling found by the algorithm will induce an optimal or close-to-optimal interpretation and set of lexical chains on the document.

Applying the Relaxation Labelling algorithm, this optimal weighted labelling $\widehat{P}$ is found, and the optimal interpretation $\widehat{\psi}$ can be obtained by taking the most highly scored label for each word variable, $\widehat{\psi}(w_i) = \mathrm{argmax}_{\sigma_j \in \phi(w_i)} \widehat{p}(w_i, \sigma_j)$.

Finally, to obtain the chains $\widehat{c}_m$, we take the sets of words in the document whose senses under the interpretation $\widehat{\psi}$ are linked by synonymy or hypernymy. Formally, we can say that w_i and $w_{i'}$ are *directly linked*, $w_i \sim w_{i'}$, iff:

$$w_i \sim w_{i'} \Longleftrightarrow \widehat{\psi}(w_i) \xrightarrow{\star} \widehat{\psi}(w_{i'}) \vee \widehat{\psi}(w_{i'}) \xrightarrow{\star} \widehat{\psi}(w_i)$$

Taking the *linked* relation, $w_i \overset{\star}{\sim} w_{i'}$, as the transitive closure of *directly linked*, the optimal lexical chaining $\widehat{C}$ is then the quotient set of the set of words in the document and the *linked* relation, $\widehat{C} = D/\overset{\star}{\sim}$, each chain $\widehat{c}_k$ being a closed set of words related through the *linked* relation.

3. Experiments

To test the validity of our approach, we have performed a preliminary set of experiments. We have used a corpus of 120 Catalan news documents coming from agency EFE, which has been also used previously for the evaluation of Automatic Summarization [21].

After the documents were processed, lexical chainings were found applying three different algorithms:

- BACKTRACK, a backtracking algorithm, warranted to find the optimal chaining, but with an exponential runtime cost.
- SILBER, the algorithm of [17], which runs in lineal time.
- RELAX, the relaxation labelling based algorithm we have presented in Section 2.3.

We considered chains of common nouns only and, following [4], we set the weight of synonymy to $t_{syn} = 10$, and the weight of hypernymy to $t_{hyp} = 4$.

For each lexical chaining produced by the algorithms, the chaining score and the runtime required to find the solution were measured. Given that the optimal lexical chaining in each document will have a different score, the *score ratio* of each solution is considered, defined as the ratio between the score of the found chaining and the score of the optimal chaining (found by BACKTRACK). Additionally, a *success rate* for each method, counted as the fraction of problems for which the found chaining equals the optimal one, is computed.

3.1. Results

Table 1 shows the average score ratio and the success rate for each one of the three considered methods, across the 120 documents in the corpus.

As we can see, for both methods the solutions found are quite close to optimal, given that the score ratios are close to 100%. RELAX only seems to slightly outperform SILBER in this aspect. However, if we take the success rate into account, it seems that RELAX is better than SILBER: whereas RELAX finds the optimal solution in 87.5% of the cases, SILBER only does so in 54.1%, a significant 33.4% below.

A Wilcoxon test between SILBER and RELAX reveals that the difference in score is significant at the 99% confidence level. Another interesting conclusion of our evaluation is that, despite the claims in [17], SILBER does not always find the optimal chaining.

	BACKTRACK	SILBER	RELAX
Score Ratio	100%	98.0%	99.6%
Success Rate	100%	54.1%	87.5%

Table 1. Average Score Ratio and Success Rate for the Considered Methods

In terms of runtime, RELAX is more expensive by a factor than SILBER (up to 4.60 in our experiments, but 1.82 in average, and with a statistically significant difference), but their asymptotic behavior is similar and close to lineal, well below the exponential runtime of BACKTRACK. Runtimes for RELAX and SILBER are small: around 25ms for the largest considered documents.

Even if these are only preliminary results, and the influence of the increase in score of the found chainings in practical applications is yet to be evaluated, at the light of these results we believe that RELAX can be successfully used for solving the Lexical Chaining problem, and that its performance is better than that of SILBER. Our intuition, as mentioned in the introduction, seems hence true: the lack of a global optimization criterion in the latter leads to non-optimal solutions, and the use of a global function optimization algorithm like the Relaxation Labelling can improve the results of Lexical Chaining.

An example of the situations where global decisions outperform local ones can be seen in Figures 2 and 3, which show the lexical chains found in a document from the evaluation corpus by the methods SILBER and RELAX, respectively. In SILBER, the greedy local strategy chooses the interpretation of *estat* (*state*) as *"the way something is with respect to its main attributes"*, and hence builds a chain with words such as *intent* (*attempt*), *forma* (*way*)... The presence of a set of words related to this sense hence justifies the local decision. However, RELAX chooses the interpretation of *estat* as *"a politically organized body of people under a single government"*, which links it to the words *reunió* (*meeting*), *país* (*country*)... and turns out to be a better decision globally. Similar changes in the chosen senses happen for *reunió*, *intent*, *forma*, *fet* (*fact*) and *seguretat* (*safety*). Ultimately, these differences lead to a significant increase in the score of the chaining found by RELAX with respect to the one found by SILBER (from 80 to 98).

Even if we expect this behavior to hold on other corpora, we believe that a more thorough evaluation on longer documents, similar to those used for the evaluation of [17] is now needed to confirm the intuitions obtained from this first evaluation.

4. Implementation Framework

As mentioned in the introduction, our Summarization system is implemented using flexible, reusable and open source Natural Language Processing components, and developed within our research group.

Linguistic Processing is performed using the FreeLing suite of language analyzers [22]. FreeLing provides a library and the required data for common language processing tasks, such as tokenization, sentence splitting, morphological analysis, POS tagging, Named Entity detection, shallow and dependency parsing, WordNet/EuroWordNet sense annotation... Currently, the Catalan, English, Galician, Italian, Spanish and Welsh lan-

Melbourne (Austràlia), 24 may (EFA).- El Gran Consell de Caps va iniciar hui, dimecres, una segona **reunió**$_1$ en què participa un representant del colpista George Speight, per a solucionar la **crisi**$_2$ política que viu el **país**$_3$ des de l'**intent**$_4$ de colp d'**estat**$_4$ del passat divendres. Tevita Vakalalabure, designat Advocat General del Govern del colpista George Speight va acudir hui al quarter en què se celebra la **reunió**$_1$ dels **caps**$_5$ natius. Es creu que la seua **participació**$_6$ té com a objectiu representar als rebels davant dels **caps**$_5$, però esta **suposició**$_7$ no ha sigut encara confirmada de **forma**$_4$ oficial. Mentrestant George Speight ha despedit el Comandant Bill del edifici del **Parlament**$_3$ on es troben els aproximadament 30 ostatges segrestats pels colpistes, després de declarar-se ahir com "**cervell**$_5$" del colp. Timoci Silatolu, vice primer ministre del **govern**$_3$ proclamat per els colpistes, va indicar a un periodista local que l'**expulsió**$_8$ de Bill es va deure al **fet**$_2$ que la seua **intervenció**$_6$ davant dels **mitjans**$_8$ de **comunicació**$_7$ posava en **perill**$_4$ la **seguretat**$_4$ del **grup**$_3$ rebel. El Comandant Bill, va dir ahir que els colpistes "matarien als rebels abans que algú intentara rescatar-los".

Chain 1:	**reunió**, **reunió**
Chain 2:	**crisi**, **fet**
Chain 3:	**país**, **Parlament**, **govern**, **grup**
Chain 4:	**intent**, **estat**, **forma**, **perill**, **seguretat**
Chain 5:	**caps**, **caps**, **cervell**
Chain 6:	**participació**, **intervenció**
Chain 7:	**suposició**, **comunicació**
Chain 8:	**expulsió**, **mitjans**

Figure 2. Sample lexical chaining obtained by SILBER (Score: 80)

Melbourne (Austràlia), 24 may (EFA).- El Gran Consell de Caps va iniciar hui, dimecres, una segona **reunió**$_1$ en què participa un representant del colpista George Speight, per a solucionar la crisi política que viu el **país**$_1$ des de l'**intent**$_2$ de colp d'**estat**$_1$ del passat divendres. Tevita Vakalalabure, designat Advocat General del Govern del colpista George Speight va acudir hui al quarter en què se celebra la **reunió**$_1$ dels **caps**$_3$ natius. Es creu que la seua **participació**$_4$ té com a objectiu representar als rebels davant dels **caps**$_3$, però esta **suposició**$_5$ no ha sigut encara confirmada de **forma**$_2$ oficial. Mentrestant George Speight ha despedit el Comandant Bill del edifici del **Parlament**$_1$ on es troben els aproximadament 30 ostatges segrestats pels colpistes, després de declarar-se ahir com "**cervell**$_3$" del colp. Timoci Silatolu, vice primer ministre del **govern**$_1$ proclamat per els colpistes, va indicar a un periodista local que l'**expulsió**$_2$ de Bill es va deure al **fet**$_2$ que la seua **intervenció**$_4$ davant dels mitjans de **comunicació**$_5$ posava en perill la **seguretat**$_5$ del **grup**$_1$ rebel. El Comandant Bill, va dir ahir que els colpistes "matarien als rebels abans que algú intentara rescatar-los".

Chain 1:	**reunió**, **país**, **estat**, **reunió**, **Parlament**, **govern**, **grup**
Chain 2:	**intent**, **forma**, **expulsió**, **fet**
Chain 3:	**caps**, **caps**, **cervell**
Chain 4:	**participació**, **intervenció**
Chain 5:	**suposició**, **comunicació**, **seguretat**

Figure 3. Sample lexical chaining obtained by RELAX (Score: 98)

guages are supported. FreeLing is distributed as open source software, under the GNU General Public License, and is publicly available[2].

Additionally, the FreeLing analyzers as well as the rest of the presented Summarization system have been integrated into a general platform for NLP [23].

The platform is based on a client/server architecture. Each linguistic processor becomes a server, whose services clients can request. Communication between servers and clients is managed by a specialized process called MetaServer, which is responsible for routing the client requests to the most suitable server and managing the sharing of these servers, as well as activating and deactivating servers as needed.

The platform is designed to allow the integration of heterogeneous components with minimum effort, as well as to reduce the coupling between clients and servers. The client requests can be routed to different servers according to user needs or to features of the data. This allows for the decomposition of Natural Language Processing systems is a set of flexible, independent and reusable components, which can be shared (statically and dynamically) between several systems. The MetaServer architecture has also been developed using free software.

Finally, we have recently developed an on-line demo of our system[3]. It is a CGI frontend script which works as a client of the MetaServer platform. After the data is processed, the script produces an HTML output from the results, which is sent back to the user.

5. Conclusions

Lexical Chains have been shown to be useful as a text representation for the location and ranking of relevant text fragments, and as a component of systems for more complex Natural Language Processing tasks, such as Automatic Summarization. However, most efficient Lexical Chaining algorithms rely on local decisions.

We have presented a formalization of the Lexical Chaining problem and a reduction to a Labelling problem, which can be solved using the Relaxation Labelling algorithm, devised for global function optimization. The results in a preliminary evaluation show that the presented approach outperforms other algorithms in the score of the found chains, with only a minor increase in runtime.

Our Lexical Chaining algorithm has been integrated on an Automatic Summarization system developed using open source reusable components, and is now publicly available via an on-line demo.

Acknowledgements

Work partially funded by the KNOW, TIN2006-15049-C03-03, and the TEXT-MESS, TIN2006-15265-C06, projects.

[2] http://garraf.epsevg.upc.es/freeling

[3] http://nidhoggr.lsi.upc.edu/~demo/flsummary-ca.html

References

[1] J. Morris, G. Hirst, *Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text*, Computational Linguistics, 17(1), 21–43, 1991.

[2] M.A. Hearst, *Multi-Paragraph Segmentation of Expository Text*, Proc. of the ACM SIGIR Conference 1994, Las Cruces, 1994.

[3] M. Okumura, T. Honda, *Word sense disambiguation and text segmentation based on lexical cohesion*, Proc. of the 15th conference on Computational Linguistics, Kyoto, Japan, Morristown, NJ, USA, 755–761, 1994.

[4] R. Barzilay, *Lexical Chains for Summarization*, Ms Thesis, Ben-Gurion University of the Negev, 1997.

[5] M. Galley, K. McKeown, *Improving Word Sense Disambiguation in Lexical Chaining*, Proc. of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03), 2003.

[6] M.A. Stairmand, *A Computational Analysis of Lexical Cohesion with applications in Information Retrieval*, PhD Thesis, UMIST, 1996.

[7] C-Y. Lin, E.H. Hovy, *Identifying Topics by Position*, Proc. of the Applied Natural Language Processing Conference (ANLP-97), Washington, DC, 1997.

[8] G. Hirst, D. St-Onge, *Lexical chains as representation of context for the detection and correction of malapropisms*, in C. Fellbaum (Ed), *WordNet: An electronic lexical database and some of its applications*, The MIT Press, Cambridge, MA, 1997.

[9] S.J. Green, *Building Hypertext Links in Newspaper Articles Using Semantic Similarity*, Department of Computer Science, University of Toronto, 1997.

[10] M.A. Hearst, *TextTiling: Segmenting Text into Multi-Paragraph Subtopic Passages*, Computational Linguistics, 23(1), 33–64, 1997.

[11] N. Stokes, *Applications of Lexical Cohesion Analysis in the Topic Detection and Tracking Domain*, PhD Thesis, University College Dublin, 2004.

[12] J. Ellman, *Using Roget's Thesaurus to Determine Similarity of Texts*, PhD Thesis, University of Sunderland, 2000.

[13] M. Mochizuki, M. Iwayama, M. Okumura, *Passage-Level Document Retrieval Using Lexical Chains*, Proc. of RIAO 2000, 2000.

[14] D. Moldovan, A. Novischi, *Lexical Chains for Question Answering*, Proc. of COLING02, Taipei, Taiwan, 2002.

[15] N. Stokes, E. Newman, J. Carthy, A.F. Smeaton, *Broadcast News Gisting using Lexical Cohesion Analysis*, Proc. of the 26th ECIR, Sunderland, U.K., 2004.

[16] J. Li, L. Sun, *A Lexical Chain Approach for Update-Style Query-Focused Multi-document Summarization*, Information Retrieval Technology, 310–320, 2008.

[17] H.G. Silber, K.F. McKoy, *Efficiently Computed Lexical Chains as an Intermediate Representation for Automatic Text Summarization*, Computational Linguistics, 28(4), pp. 487–496, 2002.

[18] L. Padró, *A Hybrid Environment for Syntax-Semantic Tagging*, PhD Thesis, Dep. Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, 1998.

[19] M. Fuentes, *A Flexible Multitask Summarizer for Documents from Different Media, Domain and Language*, PhD Thesis, Universitat Politècnica de Catalunya, 2008.

[20] C. Torras, *Relaxation and Neural Learning: Points of Convergence and Divergence*, Journal of Parallel and Distributed Computing, 6, pp. 217–244, 1989.

[21] M. Fuentes, E. Gonzàlez, H. Rodríguez, *Resumidor de Notícies en Català del Projecte Hermes*, II Congrés d'Enginyeria en Llengua Catalana, 2004.

[22] X. Carreras, I. Chao, L. Padró, M. Padró, *FreeLing: An Open-Source Suite of Language Analyzers*, Proc. of the 4th International Conference on Language Resources and Evaluation, 2004.

[23] E. Gonzàlez, *Un Sistema Genèric de Cerca de Resposta*, Degree Thesis, Facultat d'Informàtica de Barcelona, Universitat Politècnica de Catalunya, 2004.