# SeLeCT: A Lexical Cohesion Based News Story Segmentation System

Nicola Stokes [a,*], Joe Carthy [a] and Alan F. Smeaton [b] N. Stokes et al.

[a] *Department of Computer Science, University College Dublin, Ireland.*
*E-mail: {Nicola.Stokes,Joe.Carthy}@ucd.ie*
[b] *School for Computer Applications and Centre for Digital Video Processing, Dublin City University, Ireland.*
*E-mail: ASmeaton@computing.dcu.ie*

In this paper we compare the performance of three distinct approaches to lexical cohesion based text segmentation. Most work in this area has focused on the discovery of textual units that discuss subtopic structure within documents. In contrast our segmentation task requires the discovery of topical units of text i.e. distinct news stories from broadcast news programmes. Our approach to news story segmentation (the SeLeCT system) is based on an analysis of lexical cohesive strength between textual units using a linguistic technique called lexical chaining. We evaluate the relative performance of SeLeCT with respect to two other cohesion based segmenters: TextTiling and C99. Using a recently introduced evaluation metric *WindowDiff*, we contrast the segmentation accuracy of each system on both 'spoken' (CNN news transcripts) and 'written' (Reuters newswire) news story test sets extracted from the TDT1 corpus.

Keywords: Lexical Cohesion, Lexical Chaining, Text Segmentation, NLP.

## 1. Introduction

Text segmentation can be defined as the automatic identification of boundaries between distinct textual units (segments) in a piece of text. The importance and relevance of this task should not be underestimated, as good structural organisation of text is a prerequisite for many important tasks that deal with the management and presentation of data. Consider the usefulness of text segments when responding to a user query in an information retrieval task, where users are given short pieces of relevant text rather than vast quantities of semi relevant documents. Summarisation is another task that can be greatly improved by well-segmented text since the aim of this task is to identify pertinent subtopics in a document and then generate a suitable summary from this information [2].

The main motivation of our research is to investigate the usefulness of our lexical chaining technique as a means of segmenting television news programmes into distinct new stories. Lexical chaining is a linguistic technique that in general uses an auxiliary knowledge source, like the WordNet online thesaurus [12], to cluster words into sets of semantically related concepts e.g. {*motorbike, car, lorry, vehicle*}. In this paper we endeavour to explain how such constructs can be used to detect topic shifts in both CNN broadcast news programmes and Reuters newswire articles extracted from the TDT1 (Topic Detection and Tracking) corpus [1]. We define a topic shift in this context as the boundary point between two distinct news stories. In Sections 2 and 3 we discuss different text segmentation strategies. We then introduce our novel approach to lexical chain based segmentation and finally we compare the performance of our segmenter to two other lexical cohesion based segmenters; TextTiling [8] and C99 [4]. We also look at some common segmentation evaluation metrics found in the literature and compare these to the recently proposed *WindowDiff* metric [17], which aims to address a number of inadequacies with the popular $P_k$ segmentation evaluation metric [3].

*Corresponding author: Nicola.Stokes, Nicola.Stokes@ucd.ie

## 2. Text Segmentation

According to Manning [11], text segmentation techniques can be roughly separated into two different approaches; those that rely on lexical cohesion and those that rely on statistical information extraction techniques such as cue information extraction (IE). For IE techniques to work some explicit structure must be present in the text. Manning's segmenter was required to identify boundaries between real estate classified advertisements, which in general will contain the same types of information like *house price* or *location* etc. As already mentioned, research interest in the automatic detection of story boundaries in news feeds is another application of text segmentation or story segmentation as it is more commonly known, that has gained considerable momentum in recent times. Again particularly in the case of news transcripts an inherent structure exists: introduction and news summary followed by a series of news stories interspersed with commercial breaks and ending with another summary of the main news stories covered. Some researchers involved in the TDT initiative [19,3,5] have put this structure to good use by extracting cues phrases in news transcripts such as *Good Morning*, *stay with us*, *welcome back* or *reporting from PLACE* that are reliable indicators of topic shifts in the dialogue.

It has been shown that significant gains can be achieved by combining cue information with other feature information such as named entities (President Bush, George W. Bush Jr), character n-grams (sequences of word forms of length $n$), and lexical cohesion analysis [19]. These combination approaches work by learning the best indicators of segment boundaries from an annotated corpus and then combining these features in a theoretically sound framework such as a feature based language modeling approach [3], a cue based maximum entropy model [19] or a decision tree based probabilistic model [5].

One of the main problems with domain cues is that they are not only genre specific conventions used in news transcripts but they are also programme specific as well. For example in European news broadcasts, in contrast to their American counterparts, news programmes are never *brought to you by a PRODUCT NAME*. Newscaster styles also change across news stations, as certain catch phrases are favoured by some individuals more than others. The consequence of this is that new lists of cues must be generated either manually or automatically for each news sample. Hence segmenters that heavily rely on these types of cues tend to be highly sensitive to small changes in news programme structure which can have a detrimental effect on segmentation performance. A more measured approach to segmentation might use cue phrase information as secondary evidence of a topic shift and consider a domain independent technique like lexical cohesion analysis as primary evidence of the existence of a story boundary. In the following sections we look more closely at lexical cohesion as a textual characteristic and how it can be successfully used to segment text into distinct topical units. Although most story segmenters use either IE techniques, lexical cohesion techniques or a combination of both, successful results have also been achieved by using a hidden Markov modelling method more commonly used in speech recognition applications [24].

## 3. Lexical Cohesion

When reading any text it is obvious that it is not merely made up of a set of unrelated sentences, but that these sentences are in fact connected to each other in one of two ways: cohesion and coherence. Lexical cohesion is the textual quality responsible for making the sentences of a text seem 'to hang together', while coherence refers to the fact that 'there is sense in the text' [14]. Obviously coherence is a semantic relationship and needs computationally expensive processing for identification, however cohesion is a surface relationship and is hence more accessible. Cohesion can be roughly classified into three distinct classes, *reference*, *conjunction* and *lexical cohesion* [7]. Conjunction is the only class, which explicitly shows the relationship between two sentences, "James packed up their belongings *and* his father loaded them into the car". Reference and lexical cohesion on the other hand indicate sentence relationships in terms of two semantically same or related words. In the case of reference, pronouns are the most likely means of conveying referential meaning. For example in the following sentences, "*Mary* had felt unwell all day and had visited the doctor that afternoon. This explained why *she* wasn't in the mood for birthday celebrations that evening". In

order for the reader to understand that Mary is being referred to by the pronoun *her* in the second sentence, they must refer back to the first sentence. Lexical cohesion on the other hand arises from the selection of vocabulary items and the semantic relationships between them. For example, "I realised that I had a number of *books* that were overdue; the problem being that I couldn't remember where the *library* was. It turned out to be the large red brick *building* behind the town park", where cohesion is represented by the semantic relationship between the lexical items *library*, *building* and *books*. For automatic identification of these relationships it is far easier to work with lexical cohesion than reference because more underlying implicit information is needed to discover the relationship between the above pronoun and the word it references. The following examples taken from CNN news transcripts illustrate the different types of lexical cohesion that are present in text:

- **Repetition**: Occurs when a word form is repeated again in a later section of the text e.g. " In **Gaza**, though, whether the Middle East's old violent cycles continue or not, nothing will ever look quite the same once Yasir Arafat come to town. We expect him here in the **Gaza Strip** in about an hour and a half, crossing over from Egypt."
- **Repetition through synonymy**: Occurs when words share the same meaning but have two unique syntactical forms."Four years ago, it passed a domestic violence act allowing **police**, not just the victims, to press charges if they believe a domestic beating took place. In the past, **officers** were frustrated, because they'd arrive on the scene of a domestic fight, there'd be a clearly battered victim and yet, frequently, there'd be no one to file charges."
- **Word association through specialisation /generalisation**: Occurs when a specialised /generalised form of an earlier word is used. "They've put a possible **murder weapon** in O.J. Simpson's hands; that's something that no one knew before. And it shows that he bought that **knife** more than a month or two ahead of time and you might, therefore, start the theory of premeditation and deliberation."
- **Word association through part-whole /whole-part relationships**: Occurs when a part-whole /whole-part relationship exists between two words e.g. *committee* is made up of smaller parts called *members*. "The Senate Finance **Committee** has just convened. **Members** had been meeting behind closed doors throughout the morning and early afternoon."
- **Statistical associations between words**: These types of relationships occur when the nature of the association between two words cannot be defined in terms of the above relationship types. These relationships are most commonly found by word co-occurrence statistics e.g. *Osama bin Laden* and *the World Trade Centre*.

## 4. Lexical Cohesion and Text Segmentation

Research has shown that lexical cohesion is a useful device for detecting sub-topic shifts in texts [10,19,8,18,13,4], since portions of text that contain high numbers of semantically related words (cohesively strong links) generally constitute a single topical unit. So in terms of story segmentation this means that an area of low cohesive strength within a text is a good indication of a topic transition between two news stories. Most approaches to segmentation using lexical cohesion only examine patterns of syntactic repetition in the text and ignore the four other types of lexical cohesion discussed in Section 3. We will now look in detail at two such repetition based systems, since they participate in our evaluation methodology described in Section 6.

The first of these segmentation systems called TextTiling was developed by Hearst [8]. Hearst's algorithm begins by artificially fragmenting text into fixed blocks of pseudo-sentences (also of fixed length). The algorithm uses the cosine similarity metric[1] to measure cohesive strength between adjacent blocks. Depth scores are then calculated for each block based on the similarity between a block and its neighbouring blocks in the text. High values of these depth scores indicate topic boundary points as they represent areas in the text that exhibit major drops in similarity. The second cohesion based system to take part in our evaluation is

[1]The cosine similarity is often used in Information Retrieval to find the similarity between documents by measuring the cosine of the angle between two document vectors of term weights derived from the frequency of occurrence of the terms contained in each document.

Choi's segmenter C99 [4]. This is a three-step algorithm that uses image-processing techniques to interpret a graphical representation of the pair-wise similarity of each sentence in the text. The first step is to generate a sentence pair similarity matrix using the cosine similarity measure. Step two involves a technique called image ranking where each value in the similarity matrix is replaced by its rank or more specifically the proportion of neighbouring elements that have lower similarity values. The final step uses a maximization algorithm similar to Reynar's [19] to determine topic boundaries.

Another notable approach that implicitly considers all of the lexical cohesive types mentioned in Section 3 is Ponte and Crofts segmenter [18], which uses a word co-occurrence technique called LCA (Local Context Analysis) to determine the similarity between adjacent sentences. LCA works by expanding the context surrounding each sentence by finding other words and phrases that occur frequently with these sentence words in the corpus. The authors show that segmentation based on LCA is particularly suited to texts containing extremely short segments which share very few terms due to their brevity. For example, they evaluated their approach on news summaries which had an average of 2.8 sentences.

Lexical chaining based approaches to text segmentation on the other hand determine segment boundaries by analysing repetition as well as other forms of lexical cohesion in the text. There have been three previous attempts to solve text segmentation using lexical chains. The first by Okumara and Honda [15] involved an evaluation based on five Japanese texts, the second by Stairmand [21] used twelve general interest magazine articles and the third by Kan et al. [13] used fifteen *Wall Street Journal* and five *Economist* articles. In section 7, which describes our evaluation methodology, we give details of our substantially larger CNN broadcast news and Reuters newswire data sets, which represent a previously unexplored evaluation domain for lexical chain based segmentation. In the following section we introduce the SeLeCT system, a novel approach to lexical chain based segmentation that differs from other chaining attempts by the fact that we use a broader notion of lexical cohesion. More specifically, our algorithm not only examines repetition and the three basic types of cohesion (synonymy, generalization/ specialization and part-whole/ whole-part relationships) provided by the WordNet online thesaurus [12], but also a fifth form of cohesion based on statistical word association. These types of word relationships, like *demilitarisation* $\Leftrightarrow$ *Northern Ireland*, are most commonly found by generating word co-occurrence statistics, which we identify from a set of broadcast news programmes using a bi-gram model and a log-likelihood statistical association metric. This process also takes care of missing compound noun phrases in the WordNet taxonomy which are commonly used in news story descriptions like 'suicide bombing' or 'peace accord'.

## 5. SeLeCT - Segmentation using Lexical Chaining on Text

In this section we present our topic segmenter, SeLeCT. This system takes a concatenated stream of text and returns segments consisting of single news reports. The system consists of three components a 'Tokeniser', a 'Chainer' which creates lexical chains, and a 'Detector' that uses these chains to determine news story boundaries.

### *5.1. The Tokeniser*

The objective of the chain formation process is to build a set of lexical chains that capture the cohesive structure of the input stream. Before work can begin on lexical chain identification, each sample text is processed by a part-of-speech tagger. Once the nouns in the text have been identified, morphological analysis is then performed on these nouns; all plurals are transformed into their singular state, adjectives pertaining to nouns are nominalized and all sequences of words that match grammatical structures of compound noun phrases are extracted. This idea is based on a simple heuristic proposed by Justeson and Katz [9], which involves scanning part-of-speech tagged texts for patterns of adjacent tags that commonly match proper noun phrases like 'White House aid', 'PLO leader Yasir Arafat', and WordNet noun phrases like 'red wine', 'act of god', 'arms deal', and 'partner in crime'. This process also has the added advantage of removing ambiguity from the text, for example in the case of the phrase 'New York Times' which differs in meaning as a whole from the meaning of its individual

parts. In general news story proper noun phrases will not be present in WordNet, since keeping an up-to-date repository of such words is a substantial and never ending problem. However any remaining proper nouns are still useful to the chaining process since they provide a further means of capturing cohesion though repetition. One problem with compound proper noun phrases is that they are less likely to have exact syntactic repetitions elsewhere in the text. Hence we introduce into our lexical chaining algorithm a fuzzy string matcher that looks first for full syntactic match (**U.S_President→U.S_President**), then partial full-word match ($U.S$**_President→President**$\_Bush$) and finally a 'constrained' form of partial word match between the two phrases (**cave**$\_dwellers$**→cave**$rs$). In summary then, the Tokeniser produces tokenised text consisting of noun and proper noun phrases including information on their location in the text, which is then given as input to the Lexical Chainer.

### *5.2. The Lexical Chainer*

The aim of the Chainer is to find relationships between tokens (nouns, proper nouns, compound nouns, nominalized adjectives) in the data set using the WordNet thesaurus and a set of statistical word associations, and to then create lexical chains from these relationships with respect to a set of chain membership rules. The chaining procedure is based on a single-pass clustering algorithm, where the first token in the input stream forms the first lexical chain and each subsequent token is then added to an existing chain if it is related to at least one other token in that chain by any lexicographical or statistical relationships.

A stronger criterion than simple semantic similarity is imposed on the addition of a phrase to a chain, where a phrase must be added to the most recently updated and strongest[2] related chain. In addition the distance between the two tokens in the text must be less than a certain maximum number of words, depending on the strength of the relationship i.e. stronger relationships have larger distance thresholds. These system parameters are important for two reasons. Firstly these thresholds lessen the effect of spurious chains, which are weakly cohesive chains containing misidentified word associations due to the ambiguous nature of the word forms i.e. associating *gas* with *air* when *gas* refers to a *petroleum* is an example of misidentification. The creation of these sorts of chains is undesirable as they add noise to the detection of boundaries described in the next section. Secondly due to the temporal nature of news streams, stories related to important breaking-news topics will tend to occur in close proximity in time. If unlimited distance were allowed, even between strongly related words (i.e. where a repetition relationship exists), some chains would span the entire text if two stories discussing the same topic were situated at the beginning and end of a news programme.

In summary our chaining algorithm proceeds as follows, if an 'acceptable' relationship exists between a token and a chain then the token is added to that chain otherwise the token will become the seed of a new chain. This process is continued until all keywords in the text have been chained. This chaining algorithm is similar to one proposed by St Onge [20] for the detection of malapropisms in text, however statistical word associations and proper noun fuzzy matching were not considered in his implementation. Also the experiments that lead to the segmentation results discussed in Section 7 were not limited to a rigid lexical chaining style. We also investigated the effect of different combinations of lexical cohesive relationships on segmentation accuracy in order to determine optimal SeLeCT performance.

### *5.3. Boundary Detection*

The final step in the segmentation process is to pass all chain information to the boundary detector. Our boundary detection algorithm is a variation on one devised by Okumara and Honda [15] and is based on the following hypothesis: *a high concentration of chain begin and end points exist on the boundary between two distinct news stories.* We define boundary strength $w(n, n+1)$ between each paragraph in a text, as the sum[3] of the

[2]Relationship strength is ordered from strongest to weakest as follows: repetition, synonymy, generalisation/specialisation and whole-part/part-whole, and finally statistical word association.

[3]Variations of our boundary score function were experimented with e.g. product, weighted product, and weighted summation of chain begin and end point counts. The above boundary scoring function was chosen as it yielded the lowest *WindowDiff* error score (see Section 6.2). In previous work [23], we found that the product of chain begin and ends points worked best. However, these results were based on a less sophisticated prototype of the current algorithm.

number of lexical chains whose span ends at sentence $n$ and the number of chains that begin their span at sentence $n+1$. To illustrate how boundary strengths based on lexical cohesion are calculated consider the following piece of text containing one topic shift (all nouns are highlighted), accompanied by the lexical chains derived from this text fragment where chain format is:

{words... | Sentence number: chain start, chain end}

"Coming up *tomorrow* when the *hearing* resumes, we hear *testimony* from the *limousine driver* that brought O.J. Simpson to the *airport*- who brought O.J. Simpson to the *airport June* 12th, the *night* of the *murders*. The **president** of Mothers Against Drunk Driving discusses her **organization**'s support of sobriety **checkpoints** over the **holiday weekend**. She hopes **checkpoints** will be used all the **time** to limit the number of **fatalities** on the **road**."

{hearing, testimony | 1, 1}
{tomorrow, night, holiday, weekend, time | 1, 3}
{airport | 1, 1} {president, organisation | 2, 2}
{checkpoints | 2, 3} {murders, fatalities | 1, 3}

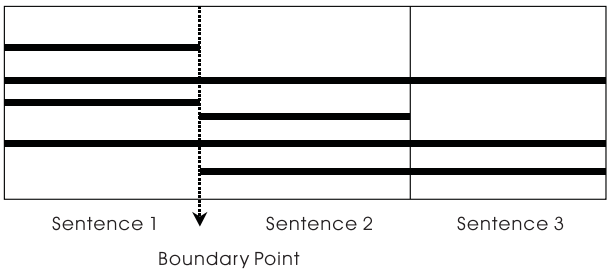


Fig. 1. Chain span schema with boundary point detected at end of sentence 1. $w(n, n+1)$ values for each of these points are $w(1,2) = (2+2) = 4$ and $w(2,3) = (1+0) = 1$.

When all boundary strengths between adjacent sentences have been calculated, as shown in Figure 1, we then get the mean of all the non-zero cohesive strength scores. This mean value then acts as the minimum allowable boundary strength that must be exceeded if the end of textual unit $n$ is to be classified as the boundary point between two news stories. Finally these boundary strength scores are cleaned using an error reduction filter. This filter removes the following common error: All boundary points which are separated by less than $x$ number of textual units, where $x$ is too small to be a 'reasonable' story length, are removed except for the boundary point with the highest score. This filter has the effect of smoothing out local maxima in the boundary score distribution, thus increasing segmentation precision. Different occurrences of this error are illustrated in Figure 2, where regions A and C represent clusters of adjacent boundary points. In this situation only the boundary with the highest score in the cluster is retained as the true story boundary. Therefore the boundary which scores 6 is retained in region A while in region C both points have the same score so in this case we consider the last point in region C to be the correct boundary position. Finally, the story boundary in region B is also eliminated because it is situated too close to the boundary points in region C and it has a lower score than either of those boundaries.

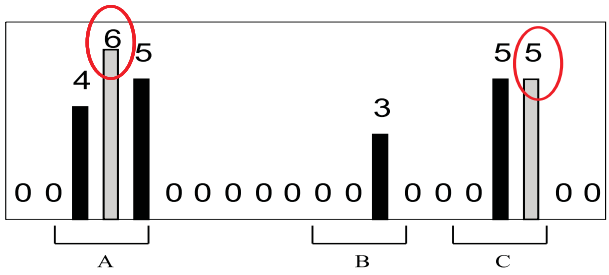


Fig. 2. Diagram shows different types of segmentation error; numbers greater than zero are possible boundary positions, while zero scores represent no story boundary point between these two textual units. Only the ringed boundaries are retained after the results are run through the error-reduction filter.

## 6. Experimental Methodology

In this section we give details of the evaluation metrics used to determine segmentation system performance in Section 7 and two news story segmentation test sets that were created from documents in the TDT1 broadcast news collection.

### 6.1. News Segmentation Test Collections

For most test collections used as input to segmentation algorithms a lot of time and effort is spent gathering human annotations i.e. human-judged topic shifts. The difficulty with these annotations lies in determining their reliability since human judges are notoriously inconsistent in their agreement on the beginning and end points of subtopic boundaries [16]. A different approach to segmentation evaluation is available to us due to the nature of the segments that we wish to detect. By concatenating distinct stories from a specific news source and using this as our test set, we eliminate subjectivity from our boundary judgments. Therefore a boundary can now be explicitly defined as the joining point between two news stories, in contrast with other test collections, which contain (disjoint) lengthy articles consisting of many subjective subtopic segments. In Section 7 we report segmentation results gathered from two test set each consisting of 1000 news stories randomly select from the TDT1 corpus. The first test set contains news stories from CNN news programme transcripts while the second contains Reuters newswire articles. Both test collections are then reorganized into 40 files each containing 25 stories. Consequently, all experimental results in Section 7 are average scores generated from individual results calculated for each of the 40 samples. After these news stories were removed from the TDT1 corpus, the remaining text was used to generate statistical word associations. As mentioned in Section 5.2, these co-occurrence relationships are then use to build lexical chains in step two of the SeLeCT segmentation process.

### 6.2. Evaluation Metrics

There has been much debate in the segmentation literature regarding appropriate evaluation metrics for estimating segmentation accuracy. Earlier experiments favoured an IR style evaluation that measures performance in terms of:

- **Recall**: The number of correctly detected story boundaries as a portion of the number of actual news boundaries in the test set.
- **Precision**: The number of correctly detected story boundaries as a portion of the total number of boundaries returned by the system.

However unlike retrieval tasks where document are classified as either relevant or non-relevant, the notion of segmentation accuracy is a fuzzier concept. For example, if a system suggests a boundary point that is one sentence away from the true story-end point it is unfair to penalize this system as heavily as a system that has missed the same boundary by 10 sentences, obviously a more fatal error. In other words recall, precision and their harmonic mean the **F1 measure** [25] all fail to take into account near-boundary misses. Consequently, these metrics are insufficiently sensitive when trying to find system parameters that yield optimal system performance [3]. Other researchers [19,18] have tried to remedy this problem by measuring recall and precision values at varying margins of error. More specifically, a system boundary is considered correct if it exists within a certain window of allowable error. So a margin of error of $+/-n$ means that if the system identifies a boundary $n$ paragraphs before or $n$ paragraphs after the correct boundary point then this end point is still counted as correct. The only stipulation is that each boundary may only be counted once as a correct boundary. This problem occurs when the value of $n$ is high and has the effect of exaggerating improvements in system performance as $n$ increases. This is the first of three metrics used in our evaluation which we define more formally as follows:

$$f_{error} = \begin{cases} 1 & \text{if } |r - s| \leq n \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

$f_{error}$ is an error function, $s$ is a system boundary point, $r$ is an actually boundary point or reference boundary and $n$ is the allowable distance in units between the actual boundary $r$ and the system boundary $s$.

Since the arrival of the TDT initiative, Beeferman et al.'s metric [3], which tries to address the inadequacies of recall and precision, has become the standard for segmentation evaluations. They proposed a **probabilistic evaluation metric** $P_k$ that aims to incorporated gradations of segmentation accuracy in terms of false positives (falsely detected segments), false negatives (missed segments) and near-misses (very close but not exact boundaries). More specifically $P_k$ is defined as 'the probability that a randomly chosen pair of words a distance $k$ words apart is inconsistently classified' [3]. However, in a recent publication Pevzner and Hearst [17] highlight several faults

with the $P_k$ metric. Most notable they criticize $P_k$ firstly for its inability to deal with different types of error in an even-handed manner and secondly for its over-sensitivity to large segment size in the test set. In the later case, $P_k$ becomes more lenient as the variance increases and in the former it unfairly penalizes false negatives more than false positives while over-penalizing near-misses. The authors show though empirical evidence and different segmentation scenarios that their proposed alternative metric called **WindowDiff** alleviates these problems and provides a fairer and more accurate performance. *WindowDiff* like $P_k$ works by moving a window of fixed size across the test set and penalizing the algorithm missed or erroneous boundary occurs. However unlike $P_k$ it calculates this error by counting 'how many discrepancies occur between the reference and the system results' rather than 'determining how often two units of text are incorrectly labelled as being in different segments' [17].

## 7. Experimental Results

In this section we present performance results for each segmenter on both the CNN and Reuters test sets with respect to the aforementioned evaluation metrics. As explained in Section 3, we determine the effectiveness of our SeLeCT system with respect to two other lexical cohesion based approaches to segmentation, namely the TextTiling [8] and C99 algorithms [4][4]. We also include results from a random segmenter that returns 25 random boundary positions for each of the 40 files in both test sets. These results represent a lower bound on segmentation performance. All results in this section are calculated using paragraphs as the basic unit of text. Since both our test sets are in SGML format, we consider the beginning of a paragraph in this context to be indicated by a speaker change tag in the CNN transcripts or a paragraph tag in the case of the Reuters news stories.

[4]We use Choi's java implementations of TextTiling and C99 available for free download at www.cs.man.ac.uk/ choif.

### *7.1. CNN Broadcast News Segmentation Results*

Table 1 summarises the results of the CNN data set for each segmentation system evaluated with respect to the three metrics. All values for these metrics range from 0 to 1 inclusively, however F1 results are expressed as 1-F1 since a score of 0, in line with the other metrics, will then represent the highest measure of system performance. Consequently, the system with the lowest score in each metric is the best performing algorithm. From the results in Table 1, we can see that the accuracy of our SeLeCT segmentation algorithm is greater than the accuracy of either C99, TextTiling or the Random segmenter for all four evaluation metrics. Although many combinations of lexical cohesive relationships were experimented with, optimal performance of the SeLeCT system was achieved when only patterns of proper noun and noun repetition were examined during the boundary detection stage. For the remainder of this subsection we will comment on the segmentation style of each of the algorithms.

The 1-F1 value for TextTiling gives us a prime example of how traditional IR metrics, precision and recall, fail as informative measures of segmentation performance. In their all-or-nothing approach to measuring segmentation performance, TextTiling rates as the worst performing system with highest overall 1-F1 score. A break down of this score shows that TextTiling's recall and precision values are very low, 27.9% and 22.4% respectively. However, these values take no account of the fact that TextTiling is producing near-misses rather than 'pure' false negatives i.e. 'just' missing boundaries rather than failing to detect them at all. To verify this we can observe from Figure 3 that recall and precision percentages significantly improves, as the margin of error is incremented in units of +/- 1 paragraph. In the case of TextTiling, this graph strongly indicates that the system is more prone to near-misses than false negatives, as the recall and precision values increase to 68.2 and 53.9 respectively at +/-1 paragraphs.

Another interesting observation from these results is that although C99 has a much lower 1-F1 measure than TextTiling in Table 1, both $P_k$ and *WindowDiff* rank it as the worst performing system. Taking a closer look at the results explains why this is the case. C99 returns nearly 3 times more 'true' false positives than TextTiling, since

| System | % Recall | % Precision | 1 – F1 | $P_k$ | WindowDiff |
|---|---|---|---|---|---|
| SeLeCT | 53.4 | 55.8 | 0.446 | 0.25 | 0.253 |
| TextTiling | 27.9 | 22.4 | 0.752 | 0.259 | 0.299 |
| C99 | 64.1 | 44.0 | 0.475 | 0.294 | 0.351 |
| Random | 7.5 | 7.5 | 0.925 | 0.421 | 0.48 |

Table 1

Precision and Recall values from segmentation on concatenated CNN news.

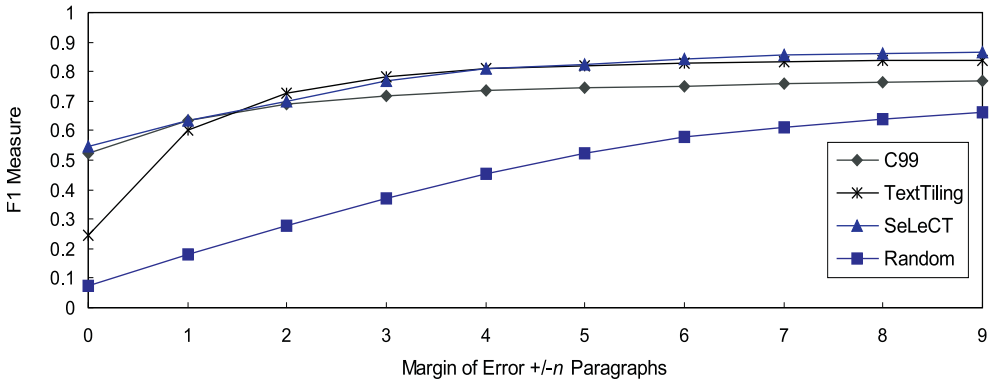


Fig. 3. Graph illustrating effects on F1 measure for each segmentation algorithm run on CNN test set as margin of error is increased.

more of TextTiling's false positives are in fact near-misses. This again is not reflected in the recall and precision values. However, Figure 3 somewhat illustrates this point by the fact that C99's performance shows the least improvement as the margin of error increases. Overall we observe that in spite of the fact that *WindowDiff* penalized each system more than $P_k$ does, the over all ranking of the systems with respect to these two measures is the same. Although, in the case of C99 and TextTiling, *WindowDiff* distinguish between their levels of accuracy with more certainty than $P_k$ does.

### *7.2. Reuters Newswire Segmentation Results*

Table 2 and Figure 4 summarise the performance of each system on our Reuters Newswire test collection. In this experiment we observe that the C99 algorithm outperforms the SeLeCT, TextTiling and Random segmenter with respect to all three evaluation metrics. Optimal performance for the SeLeCT system was once again achieved by analysing only patterns of noun phrase repetition. Overall the results show an improvement in performance for each of the systems when segmenting concatenated Reuters' news stories rather than CNN transcripts. The difference between *WindowDiff* scores (improvement in performance) for C99 is the greatest at 0.193 while SeLeCT and TextTiling exhibit less significant improvements i.e. 0.046 and 0.055 respectively. In Figure 4 we notice again the same trend in TextTiling performance which improves dramatically as the margin of error is incremented from 0 to +/-1 paragraph. This improvement is reflected in TextTiling's *WindowDiff* and $P_k$ scores which rank it a close third to the SeLeCT system. Also, we see in Table 2 as in Table 1 that although *WindowDiff* penalises system more heavily than $P_k$, the ranking of system accuracy remains the same. Pevzner and Hearst also comment on $P_k$'s sensitivity to variation in segment size in the test set. In our experiment CNN stories vary in length more than Reuters articles do. Consequently, we observe a smaller deviation between *WindowDiff* and $P_k$ scores on the Reuters collection in comparison to the CNN collection.

| System | % Recall | % Precision | 1 – F1 | $P_k$ | WindowDiff |
|---|---|---|---|---|---|
| C99 | 70 | 74.9 | 0.276 | 0.128 | 0.148 |
| SeLeCT | 60.6 | 79.1 | 0.314 | 0.191 | 0.207 |
| TextTiling | 32.1 | 41.0 | 0.640 | 0.221 | 0.244 |
| Random | 9.3 | 9.3 | 0.907 | 0.490 | 0.514 |

Table 2
Precision and Recall values from segmentation on concatenated Reuters news stories.

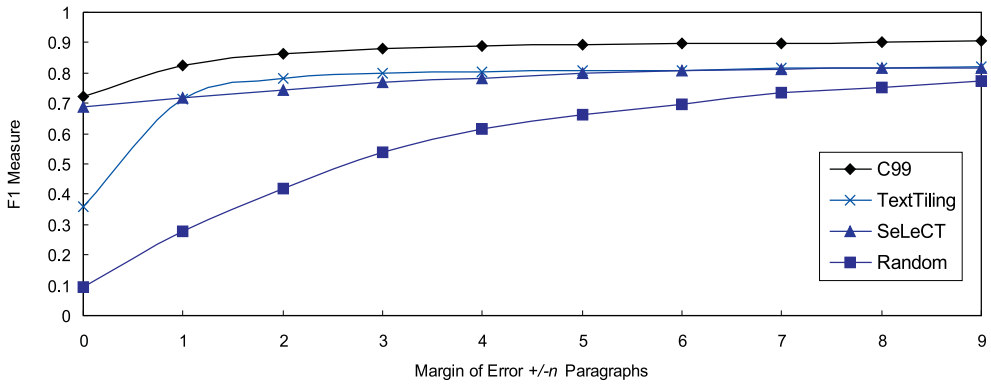


Fig. 4. Graph illustrating effects on F1 measure for each segmentation algorithm run on Reuters test set as margin of error is increased.

### *7.3. Further Comments on Results*

It is evident from our results in Sub-sections 7.1 and 7.2 that the performance of all three lexical cohesion based segmenters deteriorates when required to identify stories boundaries in 'spoken' CNN news transcripts. The most obvious explanation for lower spoken text segmentation accuracy is that transcript and news article media differ greatly in their method of conveying information. More specifically, CNN stories rely heavily on visual and audio cues like people shots or speaker change to fully convey their message. Therefore, when broadcast news is transcribed significant information loss occurs. However, resolving and incorporating these exoteric references in speech transcripts is a notoriously difficult problem.

Another crucial difference between written and spoken texts is that 'written language represents phenomena as products while spoken language represents phenomena as processes' [6]. This means that a written text usually conveys most of its meaning though nouns and adjectives, while a spoken text conveys meaning though the use of adverbs and verbs. We have found that these nominalization/verbalization trends are also evident in the CNN and Reuters collections. Obviously, this linguistic phenomenon has a significant effect on SeLeCT performance on the CNN collection, since all verbs are ignored and only patterns of noun phrases are observing during lexical chaining which results in further information loss. In [22], we show how significant improvements in SeLeCT segmentation performance can be achieved using simple heuristics which address this more manageable problem associated with spoken news transcripts.

## 8. Conclusions

In this paper we have presented a lexical chaining based approach to coarse-grained segmentation of CNN news transcripts and Reuters newswire articles. We have shown that the performance of our SeLeCT system exceeds that of the TextTiling and C99 systems when detecting topic shifts in CNN transcripts. However the results of a similar experiment on Reuters news stories showed that C99 algorithm was the best performing system.

Another interesting result reported in this paper was the failure of lexicographical and statistical relationships between words to improve segmentation accuracy, beyond that achieved by repetition based lexical chaining. Both Hearst [8] and Kan et al. [13] reported a similar conclusion; however neither of their approaches looked at statistical word associations during lexical chain formation. We believed that co-occurrence statistics would provide stronger evidence of relatedness than lexicographical relationships found in WordNet, since these associations reflect domain specific relationships between words in a news context. Nevertheless, as stated no combination of lexical cohesive relationships could match the segmentation accuracy achieved by analysing repetition based lexical chains derived from spoken or written text.

## Acknowledgements

The support of the Informatics Research Initiative of Enterprise Ireland is gratefully acknowledged. Also we wish to thank Marti Hearst for providing us with a version of the *WindowDiff* evaluation software and the reviewers for invaluable comments.

## References

[1] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang. Topic detection and tracking pilot study: Final report. In *the Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[2] R. Barzilay and M. Elhadad. Using lexical chains for text summarization. In *the Proceedings of the Intelligent Scalable Text Summarization Workshop*, 1997.

[3] D. Beeferman, A. Berger, and J. Lafferty. Statistical models for text segmentation. *Machine Learning*, 34(1-3):177–210, 1999.

[4] F. Choi. Advances in domain independent linear text segmentation. In *the Proceedings of the North American Chapter of the ACL*, 2000.

[5] S. Dharanipragada, M. Franz, J.S. McCarley, S. Roukos, and T. Ward. Story segmentation and topic detection. In *the Proceedings of the DARPA Broadcast News Workshop*, 1999.

[6] M.A.K. Halliday. *Spoken and Written Language*. Oxford University Press, 1985.

[7] M.A.K. Halliday and R. Hasan. *Cohesion in English*. Longman, 1976.

[8] M. Hearst. Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64, 1997.

[9] J.S. Justeson and S.M. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, (11):9–27, 1995.

[10] H. Kozima. Text segmentation based on similarity between words. In *the Proceedings of the Association for Computational Linguistics*, pages 286–288, 1993.

[11] Christopher D. Manning. Rethinking text segmentation models: An information extraction case study. 1998.

[12] G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Five papers on wordnet. Technical report, Cognitive Science Laboratory, 1990.

[13] Kan Min-Yen, J. L. Klavans, and K. R. McKeown. Linear segmentation and segment relevance. In *the Proceedings of the International Workshop of Very Large Corpora*, pages 197–205, 1999.

[14] J. Morris and G. Hirst. Lexical cohesion by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1), 1991.

[15] M. Okumura and T. Honda. Word sense disambiguation and text segmentation based on lexical cohesion. In *the Proceedings of the Conference on Computational Linguistics*, pages 755–761, 1994.

[16] R. Passoneau and D. Litman. Intention based segmentation: Human reliability and correlation with linguistic cues. In *the Proceedings of the Association of Computation Linguistics*, pages 148–155, 1993.

[17] L. Pevzner and M. Hearst. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36, 2002.

[18] J.M. Ponte and W.B. Croft. Text segmentation by topic. In *the Proceedings of European Conference on Digital Libraries*, pages 113–125, 1997.

[19] J. Reynar. *Topic Segmentation: Algorithms and Application*. PhD thesis, Computer and Information Science, UPenn, 1998.

[20] D. St-Onge. Detecting and correcting malapropisms with lexical chains. Master's thesis, Department of Computer Science, University of Toronto, 1995.

[21] M. A. Stairmand. *A Computational Analysis of Lexical Cohesion with Applications in Information Retrieval*. PhD thesis, Department of Language Engineering, UMIST, 1996.

[22] N. Stokes. Spoken and written news story segmentation using lexical chaining. In *the Proceedings of the Student Workshop at HLT-NAACL, Companion Volume*, pages 49–54, 2003.

[23] N. Stokes, J. Carthy, and A.F. Smeaton. Segmenting broadcast news streams using lexical chains. In *the Proceedings of STarting AI Researchers Symposium,*, pages 145–154, 2002.

[24] P. van Mulbregt, I. Carp, L. Gillick, S. A. Lowe, and J. P. Yamron. Segmentation of automatically transcribed broadcast news text. In *the Proceedings of DARPA Broadcast News Workshop*, 1999.

[25] C.J. van Rijsbergen. *Information Retrieval*. Butterworths, 1979.