Measuring Lexical Cohesion: Beyond Word Repetition

Anna Kazantseva & Stan Szpakowicz
School of Electrical Engineering and Computer Science
University of Ottawa
Ottawa, Ontario, Canada
{ankazant,szpak}@eecs.uottawa.ca

Abstract

This paper considers the problem of finding topical shifts in documents and in particular at what information can be leveraged to identify them. Recent research on topical segmentation usually assumes that topical shifts in discourse are signalled by changes in vocabulary. This information, however, is not always a sufficient indicator of a topical shift, especially for certain genres. This paper explores an additional source of information. Our hypothesis is that the type of a referring expression is an indicator of how accessible its antecedent is. The shorter and less informative the expression (*e.g.,* a personal pronoun *versus* a lengthy post-modified noun phrase), the more accessible the antecedent is likely to be and the more likely it is that the topic under discussion has remained constant between the two mentions. We explore how this information can be used to augment a lexically-based topical segmenter. We test our hypothesis on two types of data, literary narratives and lecture notes. The results suggest that our similarity metric is useful: depending on the settings it either slightly improves the performance or leaves it unchanged. They also suggest that certain types of referring expressions are more useful than others.

1 Introduction

In the past 10 years, research on topical segmentation has mostly centred on using surface vocabulary to identify topical shifts. The intuition is that if the vocabulary changes perceptibly, so does *the topic* under discussion. One popular way to model this assumption is by probabilistic graphical models. A document may be modelled as a sequence of strings (*e.g.,* sentences) generated by a latent topic variable, where the topic variables correspond to distributions over a finite vocabulary. Similarity-based methods are an alternative methodology. The segmenter explicitly measures the amount of lexical similarity between sentences. Places where similarity is low are likely to indicate shifts of topic. The common thread among these approaches is that they rely almost exclusively on the explicitly mentioned words.

The idea that vocabulary shifts indicate topical shifts dates back to Youmans (1991). Indeed, by and large, introducing new concepts almost necessarily requires that the concepts be named and described. How densely the concepts are explicitly mentioned and how often the mentions are repeated depends to a large degree on the genre and on the cognitive complexity of the document. In scientific papers or legal documents clarity is paramount, so the author will endeavour to state things explicitly and avoid ambiguity. The less complicated the document, however, the less it is necessary to explicitly repeat terminology. In literature, for example, word repetition is not only uncommon, but it is usually a sign of bad writing. In casual conversations, the topic can easily be never mentioned explicitly. How can we identify topical shifts in a document whose author does not “hold the reader’s hand”?

It turns out that lexical cohesion (or, put simply, word repetition) is only one of several devices of cohesion (Halliday and Hasan, 1976, p. 29) Other possibilities are reference, substitution, ellipsis and conjunction. In this paper we mainly explore referential cohesion.

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 476–485, Dublin, Ireland, August 23-29 2014.

Figure 1: An example dialogue from the *Moonstone* corpus

"What's wrong now?" I said once more.

"Rosanna's late again for dinner," says Nancy. "And I'm sent to fetch her in. All the hard work falls on my shoulders in this house. Let me alone, Mr. Betteredge!"

The person here mentioned as Rosanna was our second housemaid. "Where is she?" I inquired. [. . .]

"At the sands, of course!" says Nancy, with a toss of her head. "She had another of her fainting fits this morning, and she asked to go out and get a breath of fresh air. I have no patience with her!"

"Go back to your dinner, my girl," I said. "I have patience with her, and I'll fetch her in."

Figure 1 shows a snippet of a dialogue from the publicly available *Moonstone* corpus (Kazantseva and Szpakowicz, 2012). The two speakers discuss a specific person, *Rosanna*, yet her name is mentioned explicitly only twice. In the remainder of the dialogue the author uses pronouns to refer to this person, whose identity is evident from the context. Running an automatic segmenter on such a document would likely be challenging since focal concepts – characters – are often referred to by pronouns or definite noun phrases (NPs) instead of explicit repetition.

The focal entity *Rosanna* is introduced once and then it is referred to by nominal and pronominal anaphora, not by explicit repetition. Simplifying things somewhat, we can say that merely by the virtue of encountering a referring expression (*e.g., she* or *the person*), we know that it refers to something that must be clear from the context. The type of the referring expression also contains information about the availability of the antecedent. A *she* implies that the 'she' in question is rather obvious, that is to say, the antecedent is nearby and, more important for our purposes, the topical thread continues. A more verbose referring expression (*e.g., the woman in red*) is more likely in situations where the antecedent is less obvious and the reader needs additional information to disambiguate the expression.

The idea that the type of referring expression tells a lot about the accessibility of its antecedent dates back to Givón (1981). He postulated that the more informative the referring expression is, the less accessible the antecedent will be. Figure 2 shows the list of expressions from the least to the most informative. Projecting this information onto our task, we can say that the more informative the expression is, the less continuity there will be in the topic.

The main contribution of this work is to show how such information can be used to improve the quality of text segmentation. We extract NPs and classify them by informativeness. This is achieved with the help of a syntactic parser, but a lighter form of processing might do, perhaps even if it captured personal pronouns. Using this information, we augment and correct a matrix of lexical similarities between sentences, a structure frequently used as an input to a topical segmenter.

The results of using coreferential similarity are evaluated on a dataset of manually segmented chapters from a novel (Kazantseva and Szpakowicz, 2012) and on transcripts of lectures in Artificial Intelligence (Malioutov and Barzilay, 2006). We try the new similarity matrix on two publicly available similarity-based segmenters APS (Kazantseva and Szpakowicz, 2011) and MinCutSeg (Malioutov and Barzilay, 2006). The results suggest that the new matrix never hurts, and in several case improves, the performance of the segmenter, especially for the novel. We also check whether this metric would still be useful if instead of the traditionally used lexical similarity we used a similarity metric which took synonymy into account. In this case, the margin of improvement is lower, but still the coreferential similarity metric never hurts the performance and often improves it.

Section 2 of the paper gives an overview of related work. Section 3 describes our similarity metric and how we compute it. Section 4 shows the details of the experiments, while Section 5 discusses the results. We conclude in Section 6 with a discussion of how our metric can be improved and simplified.

2 Background and related work

Much of research on topical segmentation of text is based on the idea that changes of topic are usually accompanied by vocabulary changes. Introduced by Youmans (1991), it has since formed the backbone of research on topical segmentation. We now briefly review recent work on text segmentation. Since the focus of this research is on what information is useful for text segmentation, this review emphasizes representations rather than algorithms.

Perhaps the simplest way of estimating topical similarity between sentences is to measure cosine similarity between corresponding feature vectors. It has been used extensively in text segmentation. Hearst (1994; 1997) describes TextTiling, an algorithm which identifies topical shifts by sliding a window through the document and measures cosine similarity between adjacent windows. The drops in similarity signal shifts of topic. More recently, Malioutov and Barzilay (2006) as well as Kazantseva and Szpakowicz (2011) use graph cuts and factor graph clustering for text segmentation. Both systems rely on cosine similarity between bag-of-word vectors as an underlying representation.

While cosine similarity between vectors is easy to compute, it is hardly a reliable metric of topical similarity. Several researchers have used *lexical chains* – first introduced by Halliday and Hasan (1976) – to improve the performance of topical segmenters.[1] The intuition behind using lexical chains for text segmentation is that the beginning and end of a chain tend to correspond to the beginning and end of a topically cohesive segment. One version of TextTiling (Hearst, 1997) uses lexical chains manually constructed using Roget's Thesaurus. Okumura and Honda (1994) apply automatically created lexical chains to segment a small set of documents in Japanese. More recently, Marathe (2010) tried to build lexical chains using distributional semantics and apply the method to text segmentation.

Other proposals to move beyond word repetition in topical segmentation include the use of bigram overlap in (Reynar, 1999), information about collocations in (Jobbins and Evett, 1998), LSA (Landauer and Dumais, 1997) in (Choi et al., 2001; Olney and Cai, 2005) and WordNet in (Scaiano et al., 2010).

It should be noted that much of the recent work on topical segmentation revolves around generative models. For example Blei and Moreno (2001) use HMM,while Eisenstein and Barzilay (2008), Misra et al. (2011) and Du et al. (2013) use higher-order models. We do not review this work in detail here because it centers on algorithms for text segmentation and not on the information supplied to those algorithms, which is the focus of this research. Fundamentally, the text is modelled as a sequence of tokens generated by latent topic variables. Although probabilistic segmenters can be extended to use additional information (*e.g.,* Eisenstein and Barzilay (2008) augment their segmenter with information about discourse markers), it is not trivial to change these models to include information such as synonymy, co-reference and so on. That is why we do not review them in detail here.

As this brief review shows, a number of approaches have been proposed to measure cohesion between sentences, that is to say, to describe to what extent a pair of sentences is "about the same thing". Most of them have a common denominator: they use explicit lexical information, sometimes augmented by semantic relations from thesauri or ontologies.

Lexical resources, such as ontologies and knowledge-bases, may help improve the quality of segmentations, but such resources are not always available. They also may cause problems with precision. More important, however, they do not solve a more fundamental problem: a text may be highly cohesive and coherent without being tightly bound by either lexical cohesion or synonymy.

The main ideas developed in this work originate in (Givón, 1981). The author looks at the functional domain of topical accessibility. A number of coding devices affect this property. They are listed in Figure 2, ordered from the devices used to mark the most continuous topics to those which mark the least continuous topics. The order in Figure 2 is governed by a simple principle: the more accessible the topic is, the less information is used to code it. The author argues that the continuum is applicable in many languages. He also mentions that while the exact values of the phenomenon in question are difficult to predict or even estimate, their relative order can be predicted with certainty, even if some devices are unavailable in some languages.

In a similar spirit, Ariel (2014) groups non-initial NPs into expressions with *low accessibility* (definite NPs and proper names), those with *intermediate accessibility* (personal and demonstrative pronouns) and those with *high accessibility* (pronouns).

In this work, we propose to leverage the presence and type of co-referential relations to improve

[1] Very simply put, a lexical chain is a sequence of related words in a text.

the results of two recent similarity-based segmenters. Instead of resolving anaphoric references, we assume that their mere presence often indicates topic continuity. With this augmented model, we segment fiction and spoken lecture transcripts, the two types of data where low rates of lexical cohesion preclude achieving segmentation of good quality using only surface information about token types.

3 Estimating coreferential similarity

In order to see whether knowledge about types of referential expressions is useful for measuring topical similarity, we incorporate this information into two publicly available similarity-based topical segmenters, *MCSeg* (Malioutov and Barzilay, 2006) and *APS* (Kazantseva and Szpakowicz, 2011). Normally, both *MCSeg* and *APS* measure similarity between sentences by computing cosine similarity between the vectors corresponding to bag-of-words representation for each sentence:

$$sim(s_1, s_2) = \frac{s_1 \bullet s_2}{||s_1|| \times ||s_2||} \tag{1}$$

Each atomic unit of text is represented as a vector of features corresponding the occurrences of each token type. The vectors are weighted using $tf.idf$ values for each token type. Next, a segmenter measures cosine similarity between vectors according to Equation 1. That is the fundamental representation in both *MCSeg* and *APS*. *MCSeg* identifies segment boundaries by creating a weighted cyclic graph and cutting it so as to maximize the sum of edges within segments and to minimize the sum of severed edges. *APS* segments the sequence by finding segment centres – points which best capture the content of a segment – and assigning data points to best segment centres so as to maximize net similarity.

The proposed similarity metric relies on the following idea: in order to measure how many concepts two sentences share, we do not need to resolve anaphoric expressions in full, but only to map them onto sentences which contain their most recent antecedent (without actually naming the antecedents). We do that by parsing the documents with the Connexor parser (Tapanainen and Järvinen, 1997) and extracting all NPs with their constituents. Next, we attempt to classify the NPs into categories which would roughly correspond to those listed in Figure 2 and to those in (Ariel, 2014).

A manual study by Brown (1983) suggests that the average referential distance for animate and inanimate entities differs widely within the same document.[2] That is why it makes sense to distinguish between these two types. In the end, then, we classify each identified NP into one of the categories listed in Figure 3. The list is not exhaustive and in some cases an NP may belong to more than one type. In practice, however, an NP is always assigned a single type dictated by the implementation.

[2] Brown (1983, pp. 323-324) compares referring expressions which denote human and non-human entities. She uses three measurements: average distance to the nearest antecedent, average ambiguity and persistence. On all three counts, human and non-human entities appear to have different distributions.

Figure 2: Linguistic coding devices which signal topic accessibility (Givón, 1981)

Most continuous (least surprising)

1. zero anaphora
2. unstressed pronouns (*e.g., He* was speaking loudly.)
3. right-dislocated definite noun phrases (NPs) (*e.g.,* It is no good, *that book.*)
4. neutral-ordered definite NPs (*e.g., That book* is no good.)
5. left-dislocated definite NPs (*e.g., That book*, it is no good.)
6. Y-moved NP's (*e.g., The book* they read in turns.)
7. cleft/focus constructions (*e.g.,* It was *that book*, that was on her mind for weeks.)
8. referential indefinite NPs (*e.g.,* He picked up *a book* and left.)

Least continuous (most surprising)

Figure 3: Categories of noun phrases taken into account when computing coreferential similarity

1. animate personal pronouns (*he, she, they*)
2. inanimate pronouns (*it*)
3. demonstrative pronouns (*that, those*)
4. animate proper names (*John Hernecastle*)
5. inanimate proper names (*London*)
6. animate definite noun phrases (*the man*)
7. inanimate definite noun phrases (*the jewel*)
8. animate indefinite noun phrases (*a man*)
9. inanimate indefinite noun phrases (*a jewel*)

Finally, coreferential similarity between sentences S_i and S_j is measured as follows:

$$coref\_sim(S_i, S_j) = (\frac{\sum_{t=0}^{|T|} count_t^{S_j} \times weight_t}{|S_1| \times |S_2|})^{(j-i-1)\times decayFactor} \quad (2)$$

T is the set of of all types of referring expressions which we consider – those given in Figure 3. $count_t^{S_j}$ is the number of times when an expression of type t appears in the most recent sentence, S_j. Note that we only consider the referring expressions in the most recent sentence, because a referring expression, by its nature, must refer to something previously mentioned. The "tightness" of the link is controlled by setting $weight_t$ for each expression type t. $weight_t$ effectively specifies how likely it is that the antecedent for an expression of a type t appears in sentence s_i. The values of the weights are set experimentally on the holdout data. They can almost certainly be further fine-tuned. Intuitively, the settings of the weights reflect the logic behind Givón's theory. Consider an example vector of weights for expressions, where a higher weight corresponds to a more accessible antecedent (for animate and inanimate entities respectively).

<personal_pronouns_anim: 4, demonstr_pronouns_anim: 2, proper_names_anim: 1, def_np_anim: 0.5, indef_np_anim: 0, pronouns_inanim: 2, demonstr_pronouns_inanim: 2, proper_names_inanim: 0, def_np_inanim: 0, indef_np_inanim: 0>

The denominator of Equation 2 normalizes the value by the product of the lengths of sentences S_1 and S_2. The exponent $(j - i - 1) \times decayFactor$ is responsible for decreasing similarity as the distance between sentence S_i and S_j increases. The decay factor, $0 < decayFactor < 1$, is set experimentally, and $j - i$ is the distance between sentences S_i and S_j, $i < j$.

Figure 4 contains a walk-through example of computing referential similarity between two sentences.

The coreferential similarity as defined by Equation 2 is rather limited. The first limitation is the range: it can only measure similarity between nearby sentences or paragraphs, because it only makes sense between the closest occurrences of an antecedent and a subsequent referring expression. For example, it does not make sense to measure coreferential similarity between sentences that are several paragraphs apart. Even if they indeed talk about the same entities, the topic has most likely been re-introduced several times in between. That is why we only compute coreferential similarity for sentences no more than $decayWindow$ sentences apart. The value of $decayWindow$ is usually between 2 and 6 and it is set experimentally on the holdout set for each corpus.

The values of $coref_sim$ are usually quite small and the information used is rather one-sided. We use it, therefore, in addition to, not instead of, lexical similarity. In our experiments, we first compute lexical similarity between sentences (or paragraphs) and then modify the lexical matrix by adding to it the matrix of coreferential similarity.

Figure 4: An example of computing coreferential similarity

$$coref\_sim(S_i, S_j) = \left(\frac{\sum_{t=0}^{|T|} count_t^{S_j} \times weight_t}{|S_1| \times |S_2|}\right)^{(j-i-1)\times decayFactor}$$

S1: “At the sands, of course!” says Nancy, with a toss of her head.
S2: “She had another of her fainting fits this morning, and she asked to go out and get a breath of fresh air.”

Expression counts:	Weights:
personal_pronouns_anim: 2 (she, she)	4
demonstr_pronouns_anim: 0	2
proper_names_anim: 1	1
def_np_anim: 0	0.5
indef_np_anim: 0	0
pronouns_inanim: 0	2
demonstr_pronouns_inanim: 1	2
proper_names_inanim: 0	0
def_np_inanim: 2 (this morning, fainting fits)	0
indef_np_inanim: 1 (a breath)	0

$$coref\_sim(S_2, S_1) = \frac{2 \times 4 + 1 \times 1 + 1 \times 2}{21 \times 22}^{(2-1-1)\times 0.5} = 0.0234$$

4 Experimental results

The effectiveness of coreferential similarity metric has been tested in practice. A set of experiments compared how much the metric improves the quality of topical segmentations. To this end, we ran *APS* and *MCSeg* with and without adding coreferential similarity to lexical similarity, and compared the results. We chose these segmenters for comparison because $coreferential\_similarity$ can only be naturally incorporated into a similarity-based segmenter.

Data. In our experiments we used two publicly available datasets. The first one is a set of lectures on Artificial Intelligence (Malioutov and Barzilay, 2006). The dataset contains 22 documents which were manually annotated for the presence of topical shifts. The second dataset is the *Moonstone* dataset described in (Kazantseva and Szpakowicz, 2012). It contains 20 chapters from Wilkie Collins’s novel, each annotated by 4-6 people. To reconcile these multiple reference annotations, we create a majority gold standard. It only contains segment breaks which were marked by at least 30% of the annotators. Both segmenters are compared against this gold standard. There is a fair amount of disagreement between the annotators of this dataset. The average inter-annotator *windowDiff* is 0.38 (Kazantseva and Szpakowicz, 2012, pp. 215-216), but if one takes into account near-hits, then at least 50% of the boundaries are marked by more than two annotators.

Both datasets are quite challenging. The lecture dataset contain a lot of rather informal speech and there is not as much lexical repetition as would be in a more formal text. The *Moonstone* dataset is an example of literary language, full of small digressions, dialogue and so on.

The first dataset is annotated at the level of individual sentences. The second dataset is annotated at the level of paragraphs. We segment both datasets at the level of the gold-standard annotations (sentences for lectures, paragraphs for the novel).

When working with paragraphs, $coref\_sim$ is computed slightly differently:

$$coref\_sim(p_i, p_j) = \left(\frac{\sum_{t=0}^{|T|} count_t^{p_j} \times weight_t}{|p_1| \times |p_2|}\right)^{(j-i-1)\times decayFactor} \qquad (3)$$

In this case, $count_t^{p_j}$ refers to the number of occurrences of expression of type t in the first $paragraphCutOff$ sentences of the paragraph p_j, instead of the whole paragraph. The rationale behind this heuristic is that the referring expressions in the opening sentences of the paragraph are likely to refer to entities from the previous paragraph, while expressions in the middle or the end of the paragraph are likely to refer to entities introduced inside the paragraph.

Segmenters and baselines. We use two publicly available topical segmenters in our experiments: *MCSeg* and *APS*. The default version of each segmenter computes a similarity matrix between sentence in the input document. The values in the matrix correspond to cosine similarity (Equation 1) computed after the removal of stop words and weighting the bag-of-word vectors by $tf.idf$. The results obtained using these default matrices are our first baseline.

In our experiments, we modify this matrix by adding to it the matrix of coreferential similarities. The values of coreferential similarities are rather small and most modifications are localized. That is because the value of $decayWindow$ is set between 2 and 6 (see Section 3).

In addition to the matrices based on cosine similarity, we wanted to see if using a more intelligent measure of topical similarity improves the results. We built one more flavour of similarity matrices using the *DKPro Similarity* framework (Bär et al., 2013). The framework contains a model of textual similarity which has been used by the winning system at the SemEval Textual Similarity 2012 shared evaluation. We use this model (further *STS-2012*) as a more competitive baseline for computing topical similarity.

The *STS-2012* baseline consists of a log-linear regression model trained on the SemEval 2012 training data. It combines an assortment of measures of textual similarity to come up with its judgments. The metrics include n-gram overlap, semantic similarity measures (based on both corpora and lexical resources) and several measures of stylistic similarity. We chose to use this relatively complicated metric because of its competitive performance at SemEval 2012. The system, however, was not designed to measure topical similarity *per se*, especially between many sentences coming from the same source document. By default, the *STS-2012* baseline outputs values between 1 and 5. These were normalized to be between 0 and 1.

Similarly to the experimental design with cosine similarity matrices, we try running the segmenters using *STS-2012* with and without adding coreferential similarity matrix to it.

On both datasets we set the weights for various types of referential expressions using hold-out sets of two files. When setting the weights, we were guided by the principle captured in Figure 2: personal pronouns suggest the tightest link, followed by demonstrative pronouns, proper names, and so on.

It should be noted that because we had to modify the native representation of both segmenters by supplying a matrix computed using non-native code, we could not use the proper training scripts which come with the segmenters. In effect, the results are likely to be lower than they could have been. Even so, this is acceptable for our purposes because we are interested in the improvement gained by using coreferential similarity, not in obtaining the best possible segmentation via the setting of the best parameters.

Processing. We computed the underlying lexical similarity matrices using the same procedure as described in (Malioutov and Barzilay, 2006; Kazantseva and Szpakowicz, 2011), but using our own code. In other words, we built a matrix of cosine similarities after removing stop words and weighting the underlying vectors by $tf.idf$ values.

In order to compute coreferential similarity, all documents were parsed using the Connexor parser (Tapanainen and Järvinen, 1997). The parser was chosen because it produces high-quality partial parses of long sentences often encountered in the *Moonstone* dataset. We also tagged named entities and labelled NPs as animate or inanimate using the Stanford Core NLP suite.[3]

Metrics. We compare topical segmentations using the *windowDiff* metric:

$$winDiff = \frac{1}{N-k} \sum_{i=1}^{N-k} (|ref - hyp| \neq 0) \tag{4}$$

windowDiff slides a window of size k through the input sequence of length N. At every position of the window, the metric compares the number of boundaries in the reference sequence and in the hypothetical sequence. The number of erroneous windows is normalized by the total number of windows to obtain the final value. *windowDiff* is a penalty metric: lower values correspond to better segmentations.

[3] http://nlp.stanford.edu/software/corenlp.shtml

	AI Lectures	*Moonstone*
APS	0.420 (± 0.014)	0.441 (± 0.075)
APS-$coref\_sim$	0.411 (± 0.025)	0.391 (± 0.060)
APS-STS	0.428 (± 0.049)	0.479 (± 0.041)
APS-STS -$coref\_sim$	0.429 (± 0.020)	0.478 (± 0.035)
MCSeg	0.431 (± 0.045)	0.470 (± 0.095)
MCSeg-$coref_sim$	0.410 (± 0.060)	0.413 (± 0.030)
MCSeg-STS	0.451 (± 0.023)	0.441 (± 0.051)
MCSeg-STS-$coref\_sim$	0.433 (± 0.070)	0.430 (± 0.025)

Table 1: Results of comparing *APS* and *MCSeg* using four different matrix types (*windowDiff* values and standard deviation)

5 Evaluation

Table 1 presents the results of running *APS* and *MCSeg* using four different input matrices each. The first column shows the combination of the name of the segmenter and the specific input matrix. *APS* and *MCSeg* refer to the cases where both segmenters were run using simple cosine similarity matrices. STS refers to matrices computed using *STS-2012* from the *DKPro Similarity* framework. $coref\_sim$ refers to cosine similarity matrices modified by adding a matrix with coreferential similarities. $STS-coref\_sim$ are matrices computed using *STS-2012* which had coreferential similarity added to them.

In all experiments, we set the weights for different types of referring expressions on two hold-out files. The remainder of the data is divided into five folds. Standard deviation reported in the tables is computed across folds.

Coreferential similarity improves the results of the cosine matrix for both segmenters, but the improvement on the AI dataset is rather small (1% for *APS* and 2% for *MCSeg*).

It is interesting to see that in most cases using STS matrices slightly hurts the performance of the segmenters compared to using simple cosine similarity matrices. The only exception is running *MCSeg* on the *Moonstone* dataset which improves the performance by 3%.

Adding a matrix of coreferential similarities to STS matrices slightly improves the performance on the *Moonstone* dataset and leaves it practically unchanged on the dataset of AI lectures.

It is somewhat surprising that using *STS-2012* for similarity computation does not improve, and occasionally worsens, the results compared to using simple cosine similarity. Coreferential similarity, on the other hand, produces a small but consistent improvement.

We have examined the vectors of weights used in these experiments (set using hold-out data). On the *Moonstone* dataset, the results are the best when personal animate pronouns get the highest weight, followed by demonstrative animate pronouns, as well as inanimate pronouns, both regular and demonstrative. Other expression types are assigned either a very small weight or the value 0, effectively making them inconsequential. We hypothesize that this is due to the fact that the novel discusses people, their relations and interactions, making animate entities central for estimating topical links.

The vectors used on the AI lecture dataset are similar, except that here the highest weights are given to demonstrative and regular inanimate pronouns. These are followed by demonstrative and then regular animate pronouns. This distribution is likely due to the fact that the lecture dataset discusses abstract concepts, while people are likely to be noted more tangentially. We are not sure how to explain the fact that in this dataset demonstrative pronouns have a slightly higher weight than the regular ones.

Identifying and categorizing noun phrases requires either high-quality NP-tagging or parsing. On the other hand, most pronouns can be captured very easily, perhaps even using a list of words. It is interesting to note that the most gain is due to these “cheap” types of referring expressions. In the future, we plan to implement a lighter version of the coreferential similarity metric which only considers pronouns.

6 Conclusions and future work

This paper has presented a method for improving the quality of topical segmentations by using information about referential expressions in nearby sentences. The method slightly improves the quality of segmentations and, what is even more important, seems never to worsen the results.

The necessity to perform complete parsing of the input document is a drawback of the current approach. We note in Section 5, however, that the only types of referential expressions which improve performance are personal and demonstrative pronouns. Those can be easily captured without parsing. In the near future we plan to investigate such a light-weight version of $coref\_sim$ metric.

Another way to improve our current implementation would be a more objective method of setting the weights for different types of referring expressions. At present, the expressions are set by hand on a small hold-out set of documents. This is far from ideal. We plan to investigate if using logistic regression or expectation maximization would make the system more robust.

References

Mira Ariel. 2014. *Accessing Noun-Phrase Antecedents*. Routledge, London and New York.

Daniel Bär, Torsten Zesch, and Iryna Gurevych. 2013. DKPro Similarity: An Open Source Framework for Text Similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 121–126, Sofia, Bulgaria.

David Blei and Pedro Moreno. 2001. Topic segmentation with an aspect hidden Markov Model. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 343–348.

Elizabeth Brown. 1983. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022.

Freddy Y. Y. Choi, Peter Wiemer-Hastings, and Johanna Moore. 2001. Latent Semantic Analysis for Text Segmentation. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, pages 109–117, Pittsburgh, Pennsylvania.

Lan Du, Wray Buntine, and Mark Johnson. 2013. Topic Segmentation with a Structured Topic Model. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, Atlanta, Georgia.

Jacob Eisenstein and Regina Barzilay. 2008. Bayesian Unsupervised Topic Segmentation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 334–343, Honolulu, Hawaii.

Talmy Givón. 1981. Typology and Functional Domains. *Studies in Language*, 5(2):163–193.

M.A.K Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*. Longman, London and New York.

Marti A. Hearst. 1994. Multi-paragraph Segmentation of Expository Text. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, ACL '94, pages 9–16, Las Cruces, New Mexico.

Marti A. Hearst. 1997. TextTiling: segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

Amanda C. Jobbins and Lindsay J. Evett. 1998. Text Segmentation Using Reiteration and Collocation. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*, COLING '98, pages 614–618, Montréal, Québec.

Anna Kazantseva and Stan Szpakowicz. 2011. Linear Text Segmentation Using Affinity Propagation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 284–293, Edinburgh, Scotland.

Anna Kazantseva and Stan Szpakowicz. 2012. Topical Segmentation: a Study of Human Performance and a New Measure of Quality. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 211–220, Montréal, Canada.

Thomas K Landauer and Susan T. Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, pages 211–240.

Igor Malioutov and Regina Barzilay. 2006. Minimum Cut Model for Spoken Lecture Segmentation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 25–32, Sydney, Australia.

Meghana Marathe. 2010. Lexical Chains Using Distributional Measures of Concept Distance. Master's thesis, University of Toronto.

Hemant Misra, François Yvon, Olivier Cappé, and Joemon M. Jose. 2011. Text segmentation: A topic modeling perspective. *Information Processing and Management*, 47(4):528–544.

Manabu Okumura and Takeo Honda. 1994. Word Sense Disambiguation and Text Segmentation Based On Lexical Cohesion. In *COLING 1994 Volume 2: The 15th International Conference on Computational Linguistics*, pages 775–761, Kyoto, Japan.

Andrew Olney and Zhiqiang Cai. 2005. An Orthonormal Basis for Topic Segmentation in Tutorial Dialogue. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing –HLT '05*, pages 971–978, Vancouver, Canada.

Jeffrey C. Reynar. 1999. Statistical Models of Text Segmentation. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 357–364.

Martin Scaiano, Diana Inkpen, Robert Laganière, and Adele Reinhartz. 2010. Automatic Text Segmentation for Movie Subtitles. In *Advances in Artificial Intelligence*, volume 6085 of *Lecture Notes in Computer Science*, pages 295–298. Springer.

Pasi Tapanainen and Timo Järvinen. 1997. A non-projective dependency parser. *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 64–71.

Gilbert Youmans. 1991. A new tool for discourse analysis: The vocabulary-management profile. *Language*, 67(4):763–789.