Reduction of Search Space to Annotate Monolingual Corpora

Prajol Shrestha **Christine Jacquin** **Beatrice Daille**
Laboratore d'Informatique de Nantes-Atlantique (LINA)
Université de Nantes
44322 Nantes Cedex 3, France
{prajol.shrestha;christine.jacquin;beatrice.daille}@univ-nantes.fr

Abstract

Monolingual corpora which are aligned with similar text segments (paragraphs, sentences, etc.) are used to build and test a wide range of natural language processing applications. The drawback wanting to use them is the lack of publicly available annotated corpora which obligates people to make one themselves. The annotation process is a time consuming and costly task. This paper describes a new corpus-based measure to significantly reduce the search space for a faster and easier manual annotation process for monolingual corpora. This measure can be used in making alignments on different types of text segments. The performance of this measure is evaluated on a manually annotated paragraph corpus, whose alignments are freely available, with promising results.

1 Introduction

In the field of Natural Language Processing (NLP), annotated monolingual corpora are used to build and test a wide range of applications such as information retrieval, summarization, plagiarism detection, dictionary building and so on. With a number of applications to be built, the field of NLP requires a wide range of monolingual corpus with different annotations on different level of text segments. Our focus is on the fast and easy way of aligning short text segments based on similarity. These annotations are usually done manually by annotators as done by Hatzivassiloglou et al. (1999) where a corpus with alignments between similar short texts are created by two or more annotators who look at each possible short text pair independently and analyse them to make a decision on whether each pair should be aligned as similar or not. Finally, the annotators discuss the disagreements between their annotations and come to an agreement with reasoning. A corpus containing n number of short texts will generate $\frac{n(n-1)}{2}$ number of short text pairs for comparing similarities which becomes a tedious and time consuming task even if a corpus contains a few hundred of short texts. For example, the corpus we use consists of 239 paragraphs, explained in section 3.1, generating a total of 28,441 text pairs to compare.

There are few publicly available annotated corpus, some are manually annotated like the TDT corpus[1] for topic detection and tracking and the METER corpus (Gaizauskas et al., 2001) for detection of text reuse and some are automatically annotated like the PAN-PC-10 (Barrón-Cedeño et al., 2010) for plagiarism detection and the MSRPC (Dolan and Brockett, 2005) for paraphrase detection. Annotating a corpus automatically is easier and faster than manual annotation but they have a major limitation which allows the corpus to include only a subset of the problem which prevents the corpus to represent many of the naturally occurring instances. This limitation in turn might cause some incompleteness issues on the applications built on it as mentioned by Barrón-Cedeño et al. (2010) and Dolan and Brockett (2005). To reduce this effect of coverage in a corpus, annotations on corpus are done manually.

We propose manual annotation to be done in two phases. At first, the number of pairs to compare are reduced and then manual annotation is done. We present a corpus-based measure to automatically reduce the search space for manual annotation making the annotation process faster and easier. We evaluate this measure using a manually annotated paragraph corpus[2], created by reducing the search space manually.

[1] http://projects.ldc.upenn.edu/TDT-Pilot/

[2] Alignments can be freely downloaded from : http://www.projet-depart.org/public/LINA-PAL-1.0.tar.gz

Proceedings of the 5th International Joint Conference on Natural Language Processing, pages 1457–1461,
Chiang Mai, Thailand, November 8 – 13, 2011.

2 Search Space Reduction

In this section, we show how we reduce the search space manually to find similar short texts (e.g. thematic segments, paragraphs, sentences) and present a measure to automatize this manual process. Similarity is a vague concept and its definition depends on the application for which it is intended. The most general intuition of similarity is that, two short texts are similar if they have something in common (Lin, 1998). This intuition includes paraphrases, reused text, plagiarized text and so on as similar texts. Each application specifies what commonality is required to call it similar and we believe our reduction of search space will be useful for all the application based on this general intuition of similarity. This reduction of search space is the first phase towards manual annotation. This phase produces *candidate* similar pairs which is a subset of the total short text pairs within which all the actual similar pairs are present. The number of candidate similar pairs will be less than the total number of short text pairs which allows many annotators in the second phase to efficiently annotate the small set of text pairs manually in less human hours.

The manual reduction of search space is done by going through all the possible short text pairs and selecting the candidate similar pairs using a criteria which states that: each short text in a candidate similar pair consists at least one common entity (Shrestha, 2011a). This criteria for selection theoretically guarantees that all the actual similar pairs will be present in the candidate similar pairs because for two short texts to be similar they must have at least one entity in common. The entities that we use are noun, noun phrase, and transitive verb (Loberger and Shoup, 2009). Two entities are said to be common when they both have the same meaning or in other words share the same concept for example, the entities 'crashed', 'rammed into a wall', 'fatal impact' can all be mapped to the concept 'crashed' and the entities 'Prince Charles', 'heir to the British throne' can be mapped to the concept 'Prince Charles'. The context within the short text also helps to identify the concept that the entity represents.

The selection of candidate similar pairs is easier and faster because the analysis of the pairs is not required unlike when selecting actual similar pairs. This manual reduction is used while building the paragraph corpus for evaluation. As this phase is done manually, the annotator can remove a selected candidate similar pair if a decision of it not being useful can be taken easily and without any doubt to further reduce the search space.

2.1 Short text Vector Space Measure (SVSM)

We present a corpus-based measure called Short text Vector Space Measure (SVSM) (Shrestha, 2011b) based on Vector Space Model (VSM) (Salton et al., 1975) to reduce the search space. SVSM assigns a value to each text pair and text pairs having a value greater than a threshold is considered as candidate similar pairs. For simplicity reasons, we explain the method using sentences. For each sentence a sentence vector is created from term vectors. Given a corpus C of n sentences and m unique terms, the term vector, $\vec{t_j}$, for term t_j is a vector created with n number of possible dimensions where each dimension represents a unique sentence. The presence of the term in a sentence is indicated by its sentence id and the term's inverse document frequency, idf, here a document is a sentence, as shown below:

$$\vec{t_j} = [(S_1, idf_j), (S_5, idf_j), ..., (S_i, idf_j)]$$

where S_i is the sentence id where the term t_j is present, $i \in 1, .., n$ and idf_j is the idf value of term t_j. This term vector is a reduced vector space representation where sentences that do not contain the term is absent which saves space. The dimension of the matrix formed by term vectors can be further reduced using Latent Semantic Analysis (Deerwester et al., 1990) or Principle Component Analysis (Jolliffe, 1986) but are not used here. Once we have the term vectors we can create sentence vectors by adding the term vectors of the terms present in that sentence. For a sentence consisting of terms $t_1, t_2, .., t_k$, the dimension, d_i, of the sentence vector corresponding to the sentence S_i will be:

$$d_i = \Sigma_{j=1;t_j \in S_i}^{k} idf_j$$

where idf_j is the idf value of the term j and $i \in 1, .., n$. This term vector shows the different senses that the term may have. Here, the sense of the term means the idea with which it can be related to. Our assumption is that sentences are independent to each other making each sentence presenting a unique idea and therefore, each term present in a sentence is related to this idea. This assumption like the assumption of VSM (Wong et al., 1987) is unrealistic but the effect of this assumption can

-William and Harry, with their father Prince Charles and their grandmother Queen Elizabeth, are thought likely to remain in seclusion at Balmoral Castle in Scotland until Saturday's ceremony. -The royal family remained at Balmoral in Scotland Tuesday, with reports that Charles and his younger son Prince Harry went for a walk in the afternoon. It was not clear when they would return to London.
-Dodi Al Fayed's father, Harrods Department Store owner Mohammed Al Fayed, arrived here immediately after learning of his son's death. -Bernard Dartevelle, a lawyer for Mohamed Al Fayed, Dodi Fayed's wealthy businessman father and also the owner of the Hotel Ritz, said the revelation "changes absolutely nothing." He spoke of an "ambience of harassment" created around Diana and Fayed by the constant presence of paparazzi.

Table 1: Examples of similar and dissimilar paragraph pairs. The first block consists of a similar paragraph pair whereas the second block consists of a dissimilar paragraph pair.

be reduced using clustering techniques like hierarchical clustering (Han and Kamber, 2006) to group sentences that give the same idea or in other words similar sentences.

This method is similar to the method of Kaufmann (2000) using lexical cohesion but includes more information which are i) the importance of each term using its idf; ii) the co-occurrence of terms by adding up the idf value in term vectors while creating sentence vectors; and iii) the distribution of term along various sentences as the dimensions of the sentence vector is equal to the number of sentences present in the corpus. Using these sentence vectors we can now compute the similarity value between two sentences using the cosine similarity measure (Barron-Cedeno et al., 2009). In this method, other types of short text can be used in place of sentences.

3 Experiments and Results

3.1 Corpus

The corpus used for experiments was made from 12 articles on the same topic, the death of Diana, from the Linguistic Data Consortium's (LDC) North American News Text Corpus (LDC Catalog number: LDC95T21). The articles contain newswire text from three different news services which were published within two consecutive days. The articles contained 239 paragraphs, each of which contains more than 10 non stopwords, which produces 28,441 paragraph pairs for comparisons.

3.2 Manual Alignment

We have manually aligned 28,441 paragraph pairs from the corpus based on similarity. The alignment was done in two phases as explained in section 1. The first phase was performed by one annotator who selected 3,418 candidate similar paragraph pairs from a total of 28,441 paragraph pairs which took about 71 hours of work.

The second phase was done manually by two annotators who independently selected similar pairs from the candidate pairs. The similarity definition given to the annotators is an intuitive definition which states that two paragraphs are similar if one of the main information that the paragraph conveys is common. This definition is slightly different from the definition given by Shrestha (2011a) based on sub-topics. There exist few definitions on text similarity but they are all specific to the size of the text segment (Barzilay, 2003) or entities within the sentences (Hatzivassiloglou and Klavans, 2001) which make them unsuitable for a general text similarity definition. In Table 1, we present a positive and a negative example to further explain the definition. The first block presents a positive example whose main information in common is that the royal family will remain at Balmoral Castle. The paragraph pair in the second block is not similar even though the information about Dodi's father is a businessman is common because the main information conveyed by the paragraphs is different. We used kappa statistics (Carletta, 1996; Cohen, 1960) to evaluate the annotations made by the annotators in the second phase. Kappa statistics is defined as $k = \frac{P_A - P_E}{1 - P_E}$ where, in our case P_A=0.959, which is the probability of two annotators agreeing in practice and P_E=0.918, which is the expected probability of the two annotators agreeing, and k=0.5, indicating a moderate agreement (Artstein and Poesio, 2008). The error between the annotators is about 5% due to the intuitive definition of similarity. The annotators jointly resolved annotation disagreements between them by reasoning.

The second phase produced 144 similar paragraph pairs and took about 20 hours for both annotators. The total time that took to annotate the corpus manually was about 91 hours. If we had directly tried to find the actual similar paragraph

	CS		SVSM			Overlap	
T	Retri.	Rec.	Retri.	Rec.	T	Retri.	Rec.
0	**15415**	**100**	28406	100	0	**15415**	**100**
0.1	**957**	**72.92**	**17245**	**100**	1	**6618**	**96.53**
0.2	**169**	**36.11**	**7253**	**97.92**	2	**2991**	**87.5**
0.3	**51**	**17.36**	**3009**	**93.06**	3	**1434**	**77.78**
0.4	14	6.25	**1218**	**76.39**	4	735	63.89
0.5	4	2.08	412	53.47	5	398	50.69
0.6	2	1.39	134	27.78	6	197	32.64

Table 2: Rec. (Recall) and Retri. (Retrieved pairs) of methods CS, SVSM, and stem overlap according to T (Threshold).

pairs without phase one, with an assumption that the time taken per paragraph pair (≈21 sec) is the same as in the second phase, it would take about 166 hours. The total time saved is 75 hours of work.

3.3 Automatic Selection of Candidate Pairs

The manual alignment method is still time consuming and difficult as manual effort has to be done. SVSM, presented in section 2.1, is used to reduce the search space for annotators. Its performance is compared with stem overlap (Overlap) and cosine similarity measure (CS) with TF*IDF as weights (Salton and McGill, 1983). For each method, stop-words were removed and the remaining words were stemmed using the snowball stemmer[3]. We decide a paragraph pair is a candidate similar pair if the value given by a method exceeds a threshold. Table 2 shows the performance based on recall compared to the manually selected actual similar pairs of section 3.2 and the number of retrieved paragraph pairs by each method on the total paragraph pairs at different thresholds.

If we look at the table, the best result with 100% Recall is given by CS and Overlap methods with 15,415 retrieved pairs but still this is a large number. Using automatic methods, we would like to optimize our threshold so that we can reduce the retrieved paragraph pairs as much as possible without losing much of the actual similar paragraph pairs. According to the optimization issue SVSM is the best among the three methods at threshold 0.3 with 3009 retrieved paragraph pairs almost equal to the manually selected candidate pairs and with a recall of 93.06%. Another property we would like in a method for automatic reduction of search space is the slow rate of decrease in recall making sure with a small variation of threshold the recall will not have a drastic change. The rate of decrease in recall is shown in Figure 1 where four highest varying recall are plotted for each method. These values are boldfaced in Table 2. From Figure 1 we can see that SVSM is the method that has the most gradual decrease in recall making it the most suitable method for automatic reduction of search space. CS on the other hand is the least suitable with a sharp decrease in recall showing that similarity measures based only on term overlap is not suitable to find similar short text as discussed by Abdalgader (2011). Using this method at the threshold 0.3 we can reduce the time for manual annotation to about 17.5 hours (3009x21) with a loss of about 10 similar paragraph pairs only.

[3]http://snowball.tartarus.org/

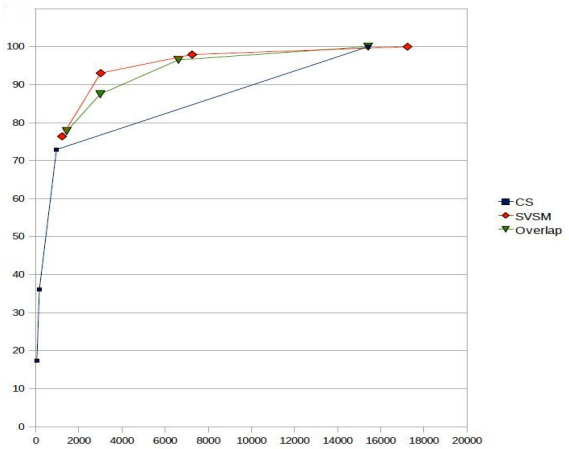

Figure 1: The rate of decrease in recall as the retrieved paragraph decreases.

4 Conclusion and Future Work

We present an automatic method using SVSM to reduce the total number of paragraph pairs from which actual similar paragraph pairs are manually selected. Using the manual method we reduced the 28,441 total paragraph comparisons to only 3,418 paragraph comparisons from which 144 paragraph pairs were aligned as similar. This shows that 99.5% of the effort in selecting the similar paragraph is wasted in terms of the difference between the end number of aligned paragraph pairs and the total initial pairs. Using the manual method we were able to save 75 hours of human work which can be further increased to 148.5 hours by using the automatic method in expense of few similar pairs. In future, the reliability of the threshold will be tested on other corpus and the present manually annotated corpus will be populated with more manually selected similar paragraph pairs using the automatic method.

Acknowledgements. This work is supported by the French Region Pays de Loire in the context of the DEPART project (http://www.projet-depart.org/).

References

Khaled Abdalgader and Andrew Skabar. 2011. Short-text similarity measurement using word sense disambiguation and synonym expansion. *AI 2010:Advances in Artificial Intelligence, Lecture Notes in Computer Science*, 6464:435–444.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Alberto Barron-Cedeno, Andreas Eiselt, and Paolo Rosso. 2009. Monoligual text similarity measures: A comparison of models over wikipedia articles revisions. *Proceedings of ICON-2009: 7th International Conference on Natural Language Processing*.

Alberto Barrón-Cedeño, Martin Potthast, Paolo Rosso, Benno Stein, and Andreas Eiselt. 2010. Corpus and Evaluation Measures for Automatic Plagiarism Detection. *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC 10)*.

Regina Barzilay. 2003. Sentence alignment for monolingual comparable corpora. In *18th Conference of the 2003 conference on Empirical methods in natural language processing*, pages 25–32.

Jean Carletta. 1996. Assessing agreement on classification tasks: The kappa statistic. In *Computational Linguistics*, pages 249–254.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. In *Educational and Psyhological Measurement*, pages 37–46.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. *3rd International Workshop on Paraphrasing (IWP2005)*.

Robert Gaizauskas, Jonathan Foster, Yorick Wilks, John Arundel, Paul Clough, and Scott Piao. 2001. The meter corpus: A corpus for analysing journalistic text reuse. pages 214–223.

Jiawei Han and Micheline Kamber. 2006. *Data Mining: Concepts and Techniques*. Number Edition, Second. Morgan Kaufmann.

Vasileios Hatzivassiloglou and Judith L. Klavans. 2001. Simfinder: A flexible clustering tool for summarization. In *Proceedings of NAACL Workshop of Automati Summarization*, pages 203–212.

Vasileios Hatzivassiloglou, Judith L. Klavans, and Eleazar Eskin. 1999. Detecting text similarity over short passages: Exploring linguistic feature combinations via machine learning. In *Proceedings of the 1999 joint sigdat conference on empirical methods in natural language processing and very large corpora*, pages 203–212.

I T Jolliffe. 1986. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):37–52.

Stefan Kaufmann. 2000. Second-order cohesion. *Computational Intelligence*, 16(4):511–524.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *ICML*, pages 296–304.

Gordon Loberger and Kate Shoup. 2009. *Websters New World English Grammar Handbook*. Wiley, Hoboken.

Gerard Salton and Michael J. McGill. 1983. *Introduction to Modern Informational Retrieval*. McGraw-Hill.

Gerard Salton, Anita Wong, and C S Yang. 1975. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Prajol Shrestha. 2011a. Alignment of monolingual corpus by reduction of the search space. In *Proceedings of the 18th Conference on the Traitement Automatique des Langues Naturelles*, volume 1, pages 543–551.

Prajol Shrestha. 2011b. Corpus-based methods for short text similarity. In *Proceedings of the 15th Rencontre des Etudiants Chercheurs en Informatique pour le Traitement automatique des Langues*, volume 2, pages 297–302.

S K M Wong, W Ziarko, V V Raghavan, and P C N Wong. 1987. On modeling of information retrieval concepts in vector spaces. *ACM Transactions on Database Systems TODS*, 12(2):299–321.