SimFinderML: Multilingual Text Similarity Identification, Experimentation, and Evaluation

David Kirk Evans
CS Department, Columbia University
devans@cs.columbia.edu

Judith L. Klavans
Center for Research on Information Access, Columbia University
klavans@cs.columbia.edu

1. INTRODUCTION

We present an initial evaluation of SimFinderML, a system for identifying similar text in multiple languages. Given a set of documents on the same topic, our goal is identify the text units that are about the same event. Our system computes a similarity metric between texts using linguistic features identified in the source language, and matched to features in the target language through standard translation mechanisms. We have implemented a rich set of complex linguistic features for English and Japanese, and are exploring support for Arabic, Chinese, and French. SimFinderML is being developed within the framework of multi-document summarization used in Columbia's NewsBlaster [3] online news browsing system.[1] This multilingual version of SimFinder will enable the incorporation of non-English text into our multi-document summaries, and is useful for summarization of multilingual digital library information.

2. SYSTEM DESCRIPTION

The concept of textual similarity is used in many applications that involve matching one text to another, such as information retrieval, categorizing texts into pre-defined categories, filtering text, or document clustering. The similarity of a document must be computed to a query, a category, a filter, or other documents. SimFinderML (Similarity Finder MultiLingual) is a program designed to identify similar texts in multilingual documents. SimFinderML is an extension of the ideas explored in SimFinder [2] developed for detecting sentence-level similarity for English text.

Our approach to multilingual text similarity identification consists of five basic steps: identifying primitives in each language, translating primitives between languages, computing feature values across primitives and translations of primitives, merging feature values into a single similarity value, and clustering text units based on the computed similarity values.

A primitive is an atomic element that is used as the basis for similarity computation. For example, one primitive we might use is the words in a sentence, while another might be dates appearing in a sentence. Sentences are then compared using the "words" and "dates" primitives. The system uses a plug-in architecture for creating primitive extractors for supported languages; there are currently nine primitive extractors implemented for English (all tokens, stemmed tokens, WordNet classes, nouns, verbs, proper nouns, heads of noun phrases, adjectives, and cardinals), ten for Japanese (tokens, five noun classes, cardinals, three verb classes), and one for Chinese (word tokens).

Once all of the primitives have been extracted from the text units, primitives from different languages are linked together by translating primitives from one language to another. Currently, a simple dictionary-based translation system is used, although we are exploring the use of more sophisticated statistical translation systems. The similarity of sentences is computed over a variety of features, which are either simple overlap metrics (e.g. the amount of overlap two sentences share on the "proper noun" primitive), or more complex composite features. Composite features are Boolean conditions specified by two primitive classes (for example, verbs and WordNet class), a window specifying the maximum distance between the two primitives, and whether the primitives must match in the same order.

The next stage merges the similarity values over multiple features between sentences to a single similarity value using a statistical model for feature merging described in [1]. The single similarity values between sentence pairs are then output, and an external clustering program is used to cluster similar sentences together.

3. ENGLISH PERFORMANCE

We have performed an evaluation of SimFinderML in English, and compared it with the previous English-only system (SimFinder). Both systems were trained on and

[1] http://newsblaster.cs.columbia.edu/

System and # of Features		Precision	Recall
SimFinder	7	23.6%	23.1%
SFML stemmed token model	1	0.71%	17.9%
SFML proper noun model	1	0.94%	20.5%
SFML large model	25	85.7%	8.95%

Table 1: Comparison of SimFinder to SimFinderML (SFML) over English data.

tested over the same data set. Eight document sets on different topics were collected from the TDT corpus,[2] and the sentences within each document set were hand labeled by judges for similarity.

Both systems were evaluated over a data set consisting of 10 articles of about 3300 words from the TDT data set on an outbreak of Ebola fever in the Democratic Republic of Congo. The precision and recall of their clustering was computed directly from the similarity judgments by assigning two sentences placed in the same cluster a "similar" label, and sentences placed in different clusters "dissimilar" labels, and comparing those labels with the human annotations.

Table 1 shows precision and recall for Simfinder and three SimFinderML runs with different models. The SimFinderML model with 25 features has 7 features on primitives (stemmed token overlap, noun overlap, verb overlap, proper noun overlap, adjectives, noun phrase heads) and a composite feature matching stemmed token and WordNet primitive pairs within a 5 word window between sentences, plus 17 additional composite features in other variations. The other two models use only a single feature as a baseline to compare against, and exhibit much lower performance. SimFinderML has a better precision than SimFinder, but worse recall; in the training run, we preferred precision to recall since our prior results showed that precision is more important than recall for our summarization application.

4. JAPANESE PERFORMANCE

We tested SimFinderML's performance on Japanese to show that the same techniques and approaches used for English are applicable to other languages. To do this, we collected three sets of articles on different topics[3] and had a native Japanese-speaking judge annotate the sentences in the article sets for similarity. Statistical models for combining feature values were generated from the training sets with 10% of the training examples held aside, to ensure some unseen cases in the test set.

Table 2 shows the precision, recall, and number of features used for four different models used for feature combination in Japanese sentence similarity identification. The baseline of random performance (choosing similar or dissimilar with equal probability) is given as a point of reference, while the first model containing only token overlap is used as a baseline, achieving 42.1% precision and 9.375% recall. The model containing only nouns achieves 52.1% precision and 13.5% recall, while a model which differentiates between proper and common nouns curiously performs worse with 42.6% precision and 7.3% recall. The best performing model contains 9 features (tokens, all nouns, all verbs, independent verbs, proper nouns, common nouns, Japanese "suru" nouns, counter affixes, and cardinals), and has a recall of 53.3% precision and 9.375% recall.

SFML Model and # of Features		Precision	Recall
random performance	–	5.02%	50.0%
tokens	1	42.1%	9.375%
nouns	1	52.1%	13.5%
proper nouns, common nouns	2	42.6%	7.3%
large model	9	53.3%	9.375%

Table 2: SimFinderML Performance for Japanese sentence similarity detection using different models.

5. FUTURE WORK

SimFinderML is a framework for testing different approaches to sentence level multilingual text similarity detection. We have shown that our approach of identifying primitives in the source language and computing similarity features over the primitives is applicable to both English and Japanese. Our next steps involve more thoroughly investigating translation models for matching primitives across languages, and investigating more complex primitives across languages. Specifically, we plan to investigate common and proper noun phrases and the effect of noun phrase variation in primitive matching.

We are also working on building a multilingual multidocument summarization system using SimFinderML. We plan on integrating it in Newsblaster for multidocument multilingual summarization of daily news, which would also prove useful for enabling information access to multilingual digital libraries.

6. REFERENCES

[1] V. Hatzivassiloglou, J. L. Klavans, M. Holcombe, R. Barzilay, M. Kan, and K. McKeown. Simfinder: A flexible clustering tool for summarization. In *NAACL'01 Automatic Summarization Workshop*, 2001.

[2] K. McKeown, J. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin. Towards multidocument summarization by reformulation: Progress and prospects. In *AAAI*, pages 453–460, 1999.

[3] K. R. McKeown, R. Barzilay, D. K. Evans, V. Hatzivassiloglou, J. L. Klavans, A. Nenkova, C. Sable, B. Schiffman, and S. Sigelman. Tracking and summarizing news on a daily basis with columbia's newsblaster. In *Proceedings of HLT 2002 Human Language Technology Conference*, San Diego, CA, 2002.

[2] http://morph.ldc.upenn.edu/Catalog/LDC98T25.html

[3] 5 articles on the February 2003 Space Shuttle Columbia explosion (63 sentences total), 5 articles on Secretary of State Colin Powell's February 6th, 2003 address to the UN (71 sentences total), and 5 articles on the Japanese government's response (44 sentences total).