Identifying Similarities and Differences Across English and Arabic News

David Kirk Evans, Kathleen R. McKeown
Department of Computer Science
Columbia University
New York, NY, 10027, USA
{devans,kathy}@cs.columbia.edu

Keywords: Multidocument multilingual summarization, text similarity, text clustering, summarization evaluation, OSINT, Foreign Language Processing

Abstract

We present a new approach for summarizing topically clustered documents from two sources, English and machine translated Arabic texts, that presents users with an overview of the differences in content of the two sources, and information that is supported by both sources. Our approach to multilingual multi-document summarization clusters all input document sentences, and identifies sentence clusters that contain information exclusive to the Arabic documents, information exclusive to the English documents, and information that is similar between the two. The result is a three-part summary describing information about the event that comes exclusively from Arabic sources, information coming exclusively from English sources, and information that both sources consider important, enabling analysts to more quickly understand differences between incoming documents from different sources. We report on a user evaluation of the summaries.

1. Introduction

Similarity has been used extensively to find important information in summarization of English news (Radev 2004, Lin&Hovy 2002, McKeown *et al.* 1999, Barzilay *et al.* 1999), but it has not been used across languages nor has the explicit identification of differences received much attention (but see Schiffman&McKeown 2004). In this paper, we present a similarity-based system, CAPS (Compare And contrast Program for Summarization), for multilingual multi-document summarization. A summary produced by CAPS identifies facts that English and Arabic sources agree on as well as explicit differences between the sources. Such a tool would be of use to an intelligence analyst assessing counter-terrorism information, political leadership or country specific information.

The approach taken in CAPS is unique in its ability to identify similarities and differences below the sentence level and to improve the quality of the summary from mixed sources over plain extraction systems by selecting English phrases to replace errorful Arabic translations.

In the following sections we first describe the CAPS architecture, then present the similarity metrics that we use for clustering and for selection of phrases for the summary. Finally, we present an evaluation of our method which quantifies both how well we identify content unique to or shared between different sources, and how well CAPS summaries capture important information. Our evaluation features the use of an automatic scoring mechanism that computes agreement in content units between a pyramid representation (Nenkova&Passonneau 2004) of the articles, separated by source. We used Arabic and English documents from the DUC 2004 multilingual corpus (Over&Yen 2004) for the experiments we report on here.

2. System Architecture

The input to CAPS are two sets of documents on an event and can be:

- a set of untranslated Arabic documents with a set of English documents, or
- a set of manual or machine translations of the Arabic documents with a set of English documents

In the experiments described in this paper, we used machine translation of the Arabic documents and English Simfinder (Hatzivassiloglou *et al.* 2001) to compute similarity.

CAPS determines similarities and differences across sources by computing a similarity metric between each pair of simplified sentences. Clustering by this metric allows the identification of all sentence fragments that say roughly the same thing. As shown in Figure 1, CAPS first simplifies the input English sentences. It does not simplify the translated Arabic sentences because these sentences are often ungrammatical and it is difficult to break them into meaningful chunks. CAPS then com-

putes similarity between each pair of simplified sentences and cluster all sentences based on the resulting values.

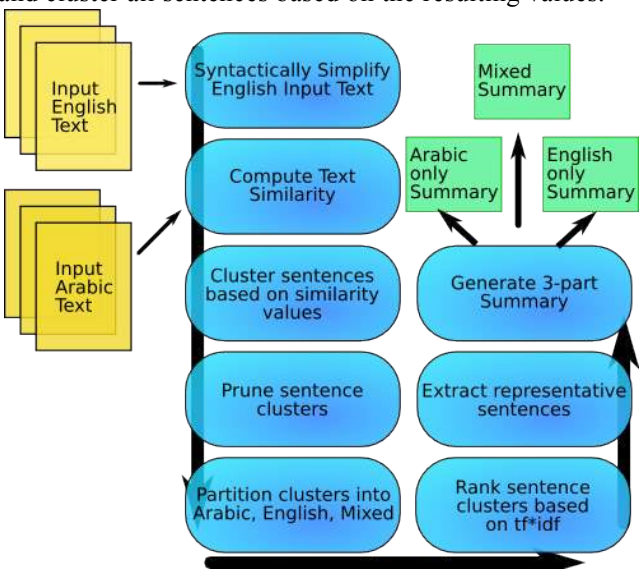

Figure 1CAPS System Architecture

Next, sentence clusters are partitioned by source, resulting in multiple clusters of similar sentences from English sources, multiple clusters of sentences from Arabic sources, and multiple clusters of sentences from both English and Arabic sources. Finally, we rank the sentences in each source partition (English, Arabic or mixed) using TF*IDF (Salton 1968); the ranking determines which clusters contribute to the summary (clusters below a threshold are not included) as well as the ordering of sentences. For each cluster, we extract a representative sentence (note that this may be only a portion of an input sentence) to form the summary. In this section, we describe each of these stages in more detail.

Sentence Simplification to Improve Clustering

Sentence simplification allows us to separate concepts that have been conveyed in a single sentence, allowing us to measure similarity at a finer grain than would otherwise be possible. We use a sentence simplification system developed at Cambridge University (Siddharthan 2002) for the task. Previous experiments with Arabic-English similarity show that we get more accurate results using simplification on the English text (Evans *et al.* 2005). The generated summary often includes only a portion of the unsimplified sentence, thus saving space and improving accuracy.

Text Similarity Computation

Text similarity between Arabic and English sentences is computed using SimFinderML, a program we developed which uses feature identification and translation at word and phrase levels to generate similarity scores. As this paper focuses on the contribution of identifying information both unique to, and similar between, the different sources, we present results using translations of the Arabic documents. The large-scale document annotation needed for the evaluation was not possible for both Arabic and English texts due to the difficulty of obtaining bilingual annotators.

Results in this paper use similarity values computed with Simfinder, an English-specific program for text similarity computation that SimFinderML was modeled after. In addition, we present a third baseline approach using the cosine distance for text similarity computation.

Sentence Clustering and Pruning

CAPS uses a non-hiearchical clustering technique, the exchange method, which casts the problem as an optimization task minimizing the intra-cluster dissimilarity (Hatzivassiloglou *et al.* 2001) over the similarity scores to produce clusters of similar sentences. Each cluster represents a fact which can be added to the summary; each sentence in the generated summary corresponds to a single cluster.

Since every sentence must be included in some cluster, individual clusters often contain some sentences that are not highly similar to others in the cluster.

To ensure that our clusters contain sentences that are truly similar, CAPS implements a cluster pruning stage that removes sentences that are not very similar to other sentences in the cluster using the same cluster pruning algorithm described in (Siddharthan *et al.* 2004). This pruning step ensures that all sentences in a sentence cluster are similar to **all other sentences** in the cluster with a similarity above a given similarity threshold.

The resulting clusters contain sentences that are much more similar to each other, which is important for our summarization strategy since we select a representative sentence from each cluster to include in the summary.

Identifying Cluster Languages

The final summary that we generate is in three parts:

- sentences available only in the Arabic documents
- sentences available only in the English documents
- sentences available in both the Arabic and English documents

After producing the sentence clusters, we partition them according to the language of the sentences in the cluster: Arabic only, English only, or Mixed. This ordering is important because it allows us to identify similar concepts across languages, and then partition them into concepts that are different: from those that are unique to the Arabic documents, and the English documents, and concepts that are supported by both Arabic and English documents. Note that these clusters are not known before-hand and are data driven, coming from the text similarity values directly.

Ranking Clusters

Once the clusters are partitioned by language, CAPS must determine which clusters are most important and should be included in the summary. Typically, there will be many more clusters than can fit in a single summary; average input data set size is 7263 words, with an average of 4050 words in clusters, and we are testing with 800 word summaries, 10% of the original text. CAPS uses

TF*IDF to rank the clusters; those clusters that contain words that are most unique to the current set of input documents are likely to present new, important information.

For each of the three types of sentence clusters, Arabic, English, and mixed, the clusters are ranked according to a TF*IDF score (Salton 1968). The TF*IDF score for a cluster is the sum of all the term frequencies in the sentences in the cluster multiplied by the inverse document frequency of the terms to discount frequently occurring terms, normalized by the number of terms in the cluster. The inverse document frequencies are computed from a large corpus of AP and Reuters news.

Sentence Selection

The cluster ranking phase determines the order in which clusters should be included in the summary. Each cluster contains several (possibly simplified) sentences, but only one of these is selected to represent the cluster in the summary. CAPS selects the sentence most similar to all other sentences in the cluster as the representative sentence for the cluster.

Only the set of unique sentences is evaluated for each cluster. Many of the input documents repeat text verbatim, as the documents are based on the same newswire (Associated Press, Reuters, etc.) report, or are updated versions of earlier reports. In order to avoid giving undue weight to a sentence that is repeated multiple times in a cluster, the unique sentences in each cluster are first identified.

To select a sentence based on the text similarity values, the average similarity of each unique sentence to every other unique sentence in the cluster is computed. The unique sentence with the highest average similarity is then chosen to represent the cluster.

In order to generate a fluent summary, CAPS draws from the English sources as much as possible. For the summary from Arabic alone, clearly we can do nothing to improve upon the machine translated Arabic. But when generating the summary from mixed English/Arabic clusters, CAPS uses English phrases in place of translated Arabic when the similarity value is above a learned threshold, a is the case for the pruned clusters. Our evaluation shows that this method improved summary quality in 68% of the cases in a human study (Evans *et al.* 2005).

Summary Generation

Once the clusters are ranked and a sentence selected to represent each cluster, the main remaining issue is how many sentences to select for each partition (English, Arabic, and mixed). There are two parameters that control summary generation: total summary word limit, and the number of sentences for each of the three partitions. The system takes sentences in proportions equal to the relative partition sizes. For example, if CAPS generates 6 Arabic clusters, 24 English clusters, and 12 mixed clusters, then the ratio of sentences from each partition is **1 Arabic : 4 English : 2 mixed**. The smallest partition size is divided through the 3 partitions to determine the ratio. The total word count is divided among partitions using this ratio.

The summary is built by extracting the number of sentences specified by the ratios computed above, and cycling continuously until the word limit has been reached. Representative sentences are chosen based on the cluster rankings computed as explained previously.

3. Evaluation

The most common method to date for evaluating summaries is to compare automatically generated summaries against model summaries written by humans for the same input set using different methods of comparison (e.g., Lin &Hovy2003, Over&Yen 2004, Radev *et al.* 2003). Since there is no corpus of model summaries that contrast differences between sources, we developed a new evaluation methodology that could answer two questions:

1. Does our approach partition the information correctly? That is, are the facts identified for inclusion in the Arabic partition actually unique to only the Arabic documents? If our similarity matching is incorrect, it may miss a match of facts across language sources.
2. Does the 3-part summary contain important information that should be included, regardless of source?

We use Summary Content Units (SCUs) (Nenkova&Passonneau 2004) to characterize the content of the documents. The Pyramid method is used to make comparisons; a pyramid weights SCUs based on how often the occur. Our evaluation features four main parts: manual annotation of all input documents and the model summaries used in DUC to identify the content units, automatic construction of four pyramids of SCUs from the annotation (one for each source and one for the entire document set), comparison of the three partitions of system identified clusters against the source specific pyramids to answer question 1 above, and comparison of the facts in the 3-part summary against the full pyramid to answer question 2.

3.1 SCU Annotation

The goal of SCU annotation is to identify sub-sentential content units that exist in the input documents. These SCUs are the facts that will serve as the basis for all comparisons. SCU annotation aims at highlighting information the documents agree on. An SCU consists of a label and contributors. The label is a concise English sentence that states the semantic meaning of the content unit. The contributors are snippet(s) of text coming from the documents or summaries that show the wording used in a specific document or summary to express the label. Each phrase of a text is part of a single SCU.

All 20 documents (10 Arabic and 10 English) and 4 summaries of 10 sets (a total of 240 documents) of the DUC data were annotated by volunteers in the Natural Language Processing group here at Columbia who are not the authors. Annotators marked SCUs in the English source and in the *manual translations* of the Arabic sources, which was available in the DUC dataset. Ma-

chine translations were too difficult for human annotators to understand.

3.2 Evaluation with SCUs

Once the SCU pyramids for a document set are created, we can use them to characterize the content of the Arabic and English documents. The SCU pyramids reveal the information in each document set, and the weights of the SCUs indicate how frequently a particular SCU was mentioned in the documents. In general, more highly weighted SCUs indicate information that we would like to include in a summary.

For example, for a set about the explosion of a Pam-Am jet over Lockerbie, Scotland, the top three SCUs from the SCU annotation broken down by language are:

Mixed Arabic and English:

- SCU 14 weight 31: The crime in question is a bombing
- SCU 17 weight 24: The bombing took place in 1988
- SCU 36 weight 22: Anan expressed optimism about the negotiations with Al Kaddafy

English only:

- SCU 57 weight 6: Libya demands the two suspects will serve time in Dutch or Libyan prisons
- SCU 121 weight 5: Libyan media reported that Al Kaddafy had no authority to hand over the two suspects
- SCU 128 weight 5: Libyan media is controlled by the government

Arabic only:

- SCU 53 weight 6: Kofi Anan informed Madeleine Albright about the discussions with Al Kaddafy
- SCU 21 weight 4: The plane involved in the bombing was an American plane
- SCU 82 weight 3: Kofi Anan visits Algeria as part of his North African tour

The SCU ID is a unique identifier for the SCU, and the weight is the number of different contributors for the SCU from all documents.

3.3 Partition evaluation

Given the system-generated ranked set of clusters for each partition (Arabic, English, mixed) we compare the SCUs found in the sentences of each cluster to the manually annotated SCUs of each language-specific pyramid. Since the SCU annotation was performed over the manual translations of the Arabic documents, identifying the SCUs in the machine translated sentences of system output was not immediate. We used a sentence alignment program to map machine translated sentences to their counterpart in the manual translation. For each system-generated sentence, the alignment program mapped the sentence to the corresponding sentence from the manual translations (which was annotated with SCUs). Using this mapping, we collected all SCUs for the representative sentence in the cluster. We then computed the percentage of these SCUs that occurred in the Arabic-only pyramid. This process was repeated for the mixed-source clusters and for the English-only cluster (although, clearly, we did not need to do alignment for the English).

We compared similarities produced by CAPS against a baseline using the cosine distance as a similarity metric.

3.4 Importance evaluation

The overall summary content quality is evaluated using the Pyramid method for summary evaluation; the full 3-part summary is scored by comparing its content to the SCU pyramid constructed for all documents in the set as well as the four human model summaries. This pyramid encodes the importance of content units in the entire set; important SCUs will appear at the top of the pyramid and will be assigned a weight that corresponds to the number of times it appears in the input documents and model summaries. The pyramid score is computed by counting each SCU present in the system generated summary, multiplied by the weight of that SCU in the gold standard pyramid. More details on pyramid scoring are available in (Nenkova&Passonneau 2004). The intuitive description of a pyramid score is that the summary receives a score ranging 0 to 1, where the score is

$$score = \frac{(Summary\ score)}{(Max\ pyramid\ score\ for\ summary)}$$

The score for the summary is simply the sum of the weights of each SCU in the summary. The max pyramid score for the summary is the maximum score one would could construct given the scoring pyramid and the number of SCUs in the summary. E.g., for a summary with 7 SCUs, the max score is the sum of the weights of the 7 biggest SCUs.

We developed an automated technique to match summary sentences to the SCUs from the pyramid. For English sentences that have been syntactically simplified, we use a longest-common-substring matching algorithm to identify the original non-simplified sentence in the annotated data. The SCUs annotated for the simplified section of the sentence are then read from the annotation data. For sentences that have not been simplified, we can read the SCUs off directly from the annotation file because they are identical.

For machine translated text, we identify the manual sentence aligned to the machine translated sentence, and read the SCUs from the annotation file that the manually translated sentence was labeled for.

Run identifier	***Arabic***	***English***	***Mixed***
Manual (CAPS)	0.2588	0.2862	0.2387
Machine (CAPS)	0.1974	0.2659	0.1195
Machine cosine	0.1909	0.0798	0.02

Table 1 Pyramid scores of representative sentence from every cluster scored against entire language pyramid

4. Results

4.1 Partition evaluation

Run Identifier	*Arabic*	*English*	*Mixed*
Manual (CAPS)	0.7748	0.7881	0.6417
Machine (CAPS)	0.7521	0.7585	0.5765
Machine cosine	0.6519	0.5377	0.3615

Table 2 Micro-averaged Pyramid scores of representative sentences from every cluster scored against corresponding language pyramid, normalized for number of SCUs.

Table 1 and Table 2 list the Pyramid scores of the three partitions using both manually translated and machine translated Arabic documents. Note that we are evaluating the representative sentence of **all clusters** in each partition, and not just the representative sentences in the summary. This evaluates how well our similarity metric clusters text for each language.

Table 1 shows the percentage of SCUs in each language pyramid that have a match in the representative sentences for the partition. The run of CAPS using manually translated Arabic documents contained sentences that covered 25.88% of the SCUs in the Arabic SCU pyramid. Given that our summaries are approximately 10% of the input text, these are perfectly acceptable recall figures. A maximal score of 1.0 would be achieved if the extracted sentence segments contained every single SCU in the pyramid. This does not happen in practice though, since not all sentences in the input documents are in the clusters; sentences that are not highly similar to other sentences are dropped. Approximately 45% of the input text does not end up in a cluster, however, almost all of the input text was annotated (although some non-relevant phrases were not annotated at the annotators' discretion.) Also, only the representative sentence is output for each cluster, and the chosen representative sentence might not contain as many SCUs as other sentences in the cluster.

The first table answers the question "how many SCUs for the language partition did we find?" while the second table answers the question "How important are the SCUs that we found?" for each language partition. For the set of clusters in each language partition we compute pyramid scores by comparison against the pyramid for that partition. Table 2 shows the micro-averaged Pyramid score normalized by the number of SCUs in the clusters for each language. The micro-average is the total weight of all cluster SCUs across all document sets divided by the total max of SCU scores across all sets. We use a micro-average instead of a macro-average (just averaging results from each set equally) because the sets are of different sizes. Micro-averaging weights large sets more than smaller sets. This normalized score indicates how important the SCUs the system covered are; a maximal score of 1.0 is achieved by choosing the highest weighted SCUs. Some SCUs are clearly less important than others, as illustrated by one of the low-weight SCUs from the Lockerbie set:

- SCU 236 weight 1: Prince Philip is the queen's husband

The run of CAPS using manually translated Arabic documents performs the best at identifying material that is exclusive to either source, and shared between the two sources. The system has particular difficulty in identifying content that is shared between the two languages, which is not surprising given the data; the annotation task was very difficult and the annotators used much world knowledge and inference in connecting the SCUs.

Using machine translated documents lowers performance, particularly in the Mixed partition. The Mixed partition is difficult because there is considerably more English text than Arabic text in the document sets, and when the machine translated Arabic text is not similar enough to the English, it is dropped from sentence clusters.

The cosine text similarity baseline performs much worse than CAPS for the English and Mixed partitions, and slightly worse for Arabic. While it covers approximately the same number of Arabic SCUs, the SCUs that it chooses are much worse, as is reflected in the micro-average pyramid score. The CAPS run with machine translated documents performs almost as well as the run with manually translated documents for the Arabic and English partitions, and only drops off for the Mixed partition.

4.2 Evaluating importance

To evaluate how well our summarizer includes important information regardless of language, we score the entire 3-part summary against the merged SCU Pyramid for each document set, and compare to two baseline systems.

The baseline systems we compare to are:

- Lead sentence extraction
- Cosine system for similarity component for clustering component

The lead sentence extraction baseline extracts the first sentence from each document until the summary length limit is reached, including the second, third, etc. sentences if there is space. The cosine baseline uses a cosine metric for text similarity computation instead of Simfinder in the CAPS framework. Table 3 shows average performance of CAPS and baseline systems over 10 different documents sets from the 2004 DUC data.

Since the pyramid sizes are different for different summaries, the average scores are computed as micro averages as before.

When using the manual translations of the Arabic documents, the CAPS system performs much better than the first sentence extraction baseline. The first sentence extraction systems perform well on this data as the first sentence of the news articles tend to include the important information from the document set that is heavily weighted in the SCU pyramid. The CAPS system, however, performs better than the first sentence extraction baseline by including a representative first sentence as well as other

Run Identifier	*Pyramid Score*
Manual Translations (CAPS)	0.8571
Manual Translations 1st sent baseline	0.7844
Machine Translations (CAPS)	0.7940
Machine Translations Cosine baseline	0.7158
Machine Translations 1st sent baseline	0.7798

Table 3 Average SCU pyramid score of CAPS and baseline systems of entire summary

sentences from sentence clusters that contain less frequently mentioned SCUs.

When using machine translations, scores are predictably lower than using manual translations; however, the CAPS system still performs better than either of the two baselines. The similarity component in CAPS performs much better than a less sophisticated text similarity technique as shown by the cosine baseline run. Interestingly, the CAPS system run over machine translated text even performs better than the first sentence extraction baseline that uses manually translated sentences.

5. Conclusions

We have presented a system for generating English summaries of a set of documents on the same event, where the documents are drawn from English and Arabic sources. Unlike previous summarization systems, CAPS explicitly identifies agreements and differences between English and Arabic sources. It uses sentence simplification and similarity scores to identify when the same facts are presented in two different sentences, and clustering to group together all sentences that report the same facts. The approach presented in the CAPS system is applicable to languages other than Arabic as long as either machine translation systems for the language pairs exist, or a multi-lingual text similarity system for the language pairs exists. We presented an evaluation methodology to measure accuracy of CAPS partitioning of similar facts by language and to score the importance of the 3-part summary content. Our evaluation shows that our similarity metric outperforms a baseline metric for identifying clusters based on language, and performs almost as well using machine translated text as manual translations for identifying important content exclusive to Arabic and English clusters.

References

Barzilay, R and McKeown, K and Elhadad, M, 1999. Information Fusion in the Context of Multi-Document Summarization. *Proceedings of the 37th Association for Computational Linguistics*. Maryland, 1999.

Evans, D.K. and McKeown, K and Klavans, J, 2005. Similarity-based Multilingual Multi-Document Summarization. *Columbia University Tech. Report* CUCS-014-05

Hatzivassiloglou, V. and Klavans, J and Holcombe, M. and Barzilay, R. and Kan, M.Y. and McKeown, K., 2001. SimFinder: A Flexible Clustering Tool for Summarization. *North American Association of Computational Linguistics 2001 Automatic Summarization Workshop.*

Lin, C.-Y. and Hovy, E.H. 2002. Automated Multi-Document Summarization in NeATS. *Proceedings of the Human Language Technology (HLT) Conference*. San Diego, CA, 2002.

Lin, C-Y and Hovy, E.H. 2003. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. *Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003)* Edmonton, Canada, May 2003.

Nenkova, A and Passonneau, R, 2004. Evaluating Content Selection in Summarization: the Pyramid Method. *Proceedings of the Human Language Technology / North American chapter of the Association for Computational Linguistics conference*. Boston, MA. May 2004.

McKeown, K. and Klavans, J. and Hatzivassiloglou, V. and Barzilay, R. and Eskin, E., 1999. Towards Multidocument Summarization by Reformulation: Progress and Prospects. *Proceedings of AAAI*, Orlando, Florida, 1999.

Over, Paul and Yen, J, 2004. An Introduction to DUC 2004 Intrinsic Evaluation of Generic News Text Summarization Systems. *NIST* http://www-nlpir.nist.gov/projects/duc/pubs/2004slides/duc2004.intro.pdf.

Radev, D. *et al.*, 2004. MEAD - a platform for multidocument multilingual text summarization. *Proceedings of LREC*. Lisbon, Portugal, May 2004.

Radev, D and Teufel, S. and Saggion, H. and Lam, W. and Blitzer, J. and Qi, H. and Elebi, A. and Liu, D. and Drabek, E., 2003. Evaluation challenges in large-scale document summarization. *Proc. of the 41st Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan, May 2003.

Salton, G, 1968. Automatic Information Organization and Retrieval. McGrawHill, New York (1968)

Schiffman, Barry and Kathleen R. McKeown, 2004. An Investigation Into the Detection of New Information. *Columbia University Technical Report CUCS-035-04*

Siddharthan, A., 2002. Resolving Attachment and Clause Boundary Ambiguities for

Simplifying Relative Clause Constructs. *Proceedings of the Student Workshop, 40th Meeting of the Association for Computational Linguistics (ACL'02).* Philadelphia, USA, 60-65.

Siddharthan, A. and Nenkova, A. and McKeown, K., 2004. Syntactic Simplification for Improving Content Selection in Multi-Document Summarization. *Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004).* Geneva, Switzerland