

Contents

1	Introduction	4
1.1	Motivation	4
1.2	Research Question and Goals	4
2	Romansh	6
2.1	Rhaeto-Romance	6
2.2	Romansh	7
2.3	Rumantsch Grischun	8
2.3.1	Lia Rumantscha	8
2.3.2	Rumantsch Grischun	8
2.3.3	Properties	8
2.3.4	Today	9
3	Compiling the Corpus	10
3.1	Introduction	10
3.2	Collecting the Data	10
3.3	Web Scraping	10
3.4	Building the Corpus	12
3.4.1	HTML Parsing	12
3.4.2	Document Alignment	13
3.5	Manual alignment of unlinked documents	16
3.6	SQLite database	16
3.7	Summary	16
4	Sentence Alignment	18
4.1	Introduction	18
4.1.1	Formal definition	18
4.2	Method Overview	19
4.2.1	Length Based	19
4.2.2	Partial Similarity Based	19
4.2.3	Translation based	20
4.2.4	Hybrid models	20
4.2.5	Summary	20
4.3	More Recent methods	21

4.3.1	Bleualign	21
4.3.2	Vecalign	22
4.4	Sentence alignment pipeline	22
4.4.1	Tool of choice	22
4.4.2	Pipeline	23
4.4.3	Sentence segmentation	23
4.4.4	Aligning language pairs	24
4.4.5	Filtering and tokenizing	25
4.5	Results	25
5	Word Alignment	27
5.1	Introduction	27
5.2	Overview of Methods	28
5.2.1	IBM Model 1	28
5.2.2	Higher IBM Models	29
5.3	Word Embeddings	30
5.3.1	Excursion: Words	30
5.3.2	Word Embeddings	31
5.3.3	Word Similarity	32
5.3.4	Multilingual Word Embeddings	32
5.3.5	Summary	33
5.4	Similarity Based Word Alignment	33
5.4.1	Method	33
5.4.2	Summary	35
6	Gold standard	36
6.1	Introduction	36
6.2	Sure and Possible Alignments	36
6.3	Evaluation Metrics	37
6.4	Gold standard for German-Romansh	37
6.4.1	Annotation tool	37
6.4.2	Guidelines	38
6.4.3	General principles	38
6.4.4	Examples	39
6.5	Flaws	41
7	Results	43
7.1	Baseline System	43
8	Summary	44
	List of Tables	46
	List of Figures	47

Chapter 7

Results

After having created a gold standard (see Chapter 6) for evaluating the quality of the alignments, I compared the alignments computed by SimAlign with the alignments computed by a baseline system. I shall now proceed present the results of the experiment.

7.1 Baseline System

As a baseline system, I chose `fast_align` (Dyer, Chahuneau, and Smith 2013). `fast_align` is a re-parameterization of the IBM Model 2. It has become a popular seccessor to Giza++, serves as a baseline system in other works, and is even recommended by WHO? as an alternative for Giza++ for computing the word alignments for Moses SMT. It outperforms Giza++ in many scenarios.

`fast_align` is extremely fast—computing the word alignments for the around 80,000 sentence pairs took around 50 seconds. It is well documented and is extremely easy to compile and to operate. All of this makes `fast_align` the most attractive system to use as a baseline system.

	Dataset size	Percision	Recall	F1	AER
Baseline	~80k	0.625	0.786	0.696	0.304
	50k	blah	blah	blah	blah
	25k				
	10k				
	5k				
	600				

Glossary

Graubünden The Canton of Grisons. 7

Acronyms

AER Average Error Rate. 35

List of Tables

2.1	Examples for choosing the forms for Rumanstch Grischun, based on liver1999 . .	9
4.1	Parallel corpus in numbers	25
6.1	Translation examples of German compounds into Romansh	39

List of Figures

2.1	Distribution of Rhaeto-Romance, taken from haiman1992	7
3.1	Directory tree of corpus_builder	11
3.2	Directory scheme for saving the HTML files	11
3.3	Portion of automatically aligned press releases up to 2009	14
4.1	Sentence alignment pipeline	22
5.1	Word alignment example	27
5.2	Similarity matrix	34
5.3	Alignment matrix	34
5.4	The resulting word alignment	34
6.1	Aligning German perfect to Romansh perfect	40
6.2	Alignment of German preterite to Romansh perfect	40
6.3	Aligning German present participles to Romansh relative clauses	41

Bibliography

Dyer, Chris, Victor Chahuneau, and Noah A. Smith (June 2013). “A Simple, Fast, and Effective Reparameterization of IBM Model 2”. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Atlanta, Georgia: Association for Computational Linguistics, pp. 644–648. URL: <https://aclanthology.org/N13-1073>.