A hybrid approach to align sentences and words in English-Hindi parallel corpora

Niraj Aswani
Department of Computer Science
University of Sheffield
Regent Court 211, Portobello Street
Sheffield S1 4DP, UK
N.Aswani@dcs.shef.ac.uk

Robert Gaizauskas
Department of Computer Science
University of Sheffield
Regent Court 211, Portobello Street
Sheffield S1 4DP, UK
R.Gaizauskas@dcs.shef.ac.uk

Abstract

In this paper we describe an alignment system that aligns English-Hindi texts at the sentence and word level in parallel corpora. We describe a simple sentence length approach to sentence alignment and a hybrid, multi-feature approach to perform word alignment. We use regression techniques in order to learn parameters which characterise the relationship between the lengths of two sentences in parallel text. We use a multi-feature approach with dictionary lookup as a primary technique and other methods such as local word grouping, transliteration similarity (edit-distance) and a nearest aligned neighbours approach to deal with many-to-many word alignment. Our experiments are based on the EMILLE (Enabling Minority Language Engineering) corpus. We obtained 99.09% accuracy for many-to-many sentence alignment and 77% precision and 67.79% recall for many-to-many word alignment.

1 Introduction

Text alignment is not only used for the tasks such as bilingual lexicography or machine translation but also in other language processing applications such as multilingual information retrieval and word sense disambiguation. Whilst resources like bilingual dictionaries and parallel grammars help to improve Machine Translation (MT) quality, text alignment, by aligning two texts at various levels (i.e. documents, sections, paragraphs, sentences and words), helps in the creation of such lexical resources (Manning & Schütze, 2003).

In this paper, we describe a system that aligns English-Hindi texts at the sentence and word level. Our system is motivated by the desire to develop for the research community an alignment system for the English and Hindi languages. Building on this, alignment results can be used in the creation of other Hindi language processing resources (e.g. part-of-speech taggers). We present a simple sentence length approach to align English-Hindi sentences and a hybrid approach with local word grouping and dictionary lookup as the primary techniques to align words.

2 Sentence Alignment

Sentence alignment techniques vary from simple character-length or word-length techniques to more sophisticated techniques which involve lexical constraints and correlations or even cognates (Wu 2000). Examples of such alignment techniques are Brown et al. (1991), Kay and Roscheisen (1993), Warwick et al. (1989), and the "align" programme by Gale and Church (1993).

2.1 Length-based methods

Length-based approaches are computationally better, while lexical methods are more resource

Proceedings of the ACL Workshop on Building and Using Parallel Texts, pages 57–64,
Ann Arbor, June 2005. ©Association for Computational Linguistics, 2005

hungry. Brown et al. (1991) and Gale and Church (1993) are amongst the most cited works in text alignment work. Purely length-based techniques have no concern with word identity or meaning and as such are considered knowledge-poor approaches. The method used by Brown et al. (1991) measures sentence length in number of words. Their approach is based on matching sentences with the nearest length. Gale and Church (1993) used a similar algorithm, but measured sentence length in number of characters. Their method performed well on the Union Bank of Switzerland (UBS) corpus giving a 2% error rate for 1:1 alignment.

2.2 Lexical methods

Moving towards knowledge-rich methods, lexical information can be vital in cases where a string with the same length appears in two languages. Kay and Roscheisen (1993) tried lexical methods for sentence alignment. In their algorithm, they consider the most reliable pair of source and target sentences, i.e. those that contain many possible lexical correspondences. They achieved 96% coverage on Scientific American articles after four passes of the algorithm. Other examples of lexical methods are Warwick et al. (1989), Mayers et al. (1998), Chen (1993) and Haruno and Yamazaki (1996).

Warwick et al. (1989) calculate the probability of word pairings on the basis of frequency of source word and the number of possible translations appearing in target segments. They suggest using a bilingual dictionary to build word-pairs. Mayers et al. (1998) propose a method that is based on a machine readable dictionary. Since bilingual dictionaries contain base forms, they pre-process the text to find the base form for each word. They tried this method in an English-Japanese alignment system and got accuracy of about 89.5% for 1-to-1 and 42.9% for 2-to-1 sentence alignments. Chen (1993) constructs a simple word-to-word translation model and then takes the alignment that maximizes the likelihood of generating the corpus given the translation model. Haruno and Yamazaki (1996) use a POS tagger for source and target languages and use an online dictionary to find matching word pairs. Haruno and Yamazaki (1996) pointed out that though dictionaries cannot capture context dependent keywords in the corpus, they can be very useful to obtain information about words that appear only once in the corpus. Lexical methods for sentence alignment may also result in partial word alignment. Given that lexical methods can be computationally expensive, our idea was to try a simple length-based approach similar to that of Brown et al. (1991) for sentence alignment and then use lexical methods to align words within aligned sentences.

2.3 Algorithm

We use English-Hindi parallel data from the EMILLE corpus for our experiments. EMILLE is a 63 Million word electronic corpus of South Asian languages, especially those spoken as minority languages in UK. It has around 120,000 words of parallel data in each of English, Hindi, Urdu, Punjabi, Bengali, Gujarati, Sinhala and Tamil (Baker et al., 2004).

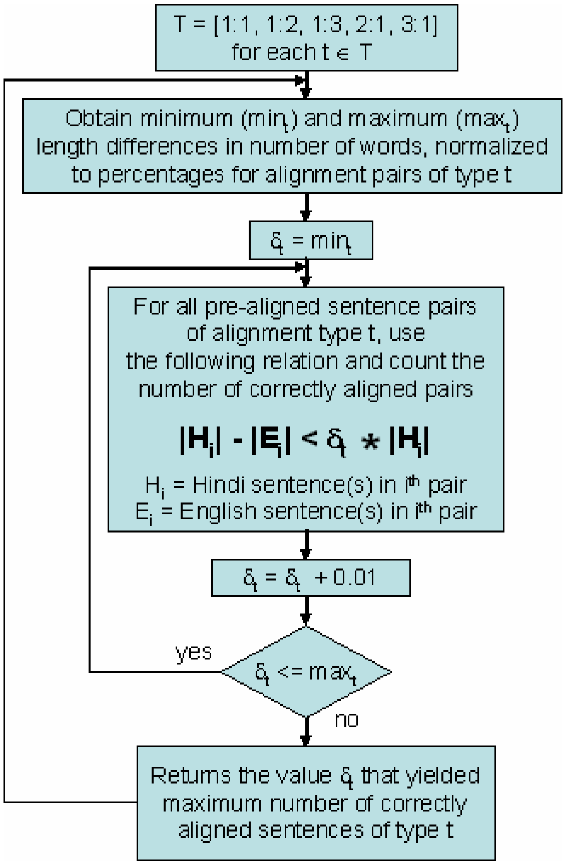

T = [1:1, 1:2, 1:3, 2:1, 3:1]
for each t ∈ T
Obtain minimum (min_t) and maximum (max_t) length differences in number of words, normalized to percentages for alignment pairs of type t
$\delta_t = min_t$
For all pre-aligned sentence pairs of alignment type t, use the following relation and count the number of correctly aligned pairs
$|H_i| - |E_i| < \delta_t * |H_i|$
H_i = Hindi sentence(s) in i^{th} pair
E_i = English sentence(s) in i^{th} pair
$\delta_t = \delta_t + 0.01$
yes
$\delta_t <= max_t$
no
Returns the value δ_t that yielded maximum number of correctly aligned sentences of type t

Figure 2.1 Sentence Alignment Parameter Learning algorithm

Table 2.1 Rules for the Sentence Alignment Algorithm

Rule	If	Hindi:English Alignment
H1	$\lvert h_i \rvert - (\lvert e_j \rvert + \lvert e_{j+1} \rvert) < 0.17 * \lvert h_i \rvert$	1-To-2
H2	$\lvert h_i \rvert - (\lvert e_j \rvert + \lvert e_{j+1} \rvert + \lvert e_{j+2} \rvert) < 0.17 * \lvert h_i \rvert$	1-To-3
E1	$\lvert e_j \rvert - (\lvert h_i \rvert + \lvert h_{i+1} \rvert) < 0.17 * \lvert e_j \rvert$	2-To-1
E2	$\lvert e_j \rvert - (\lvert h_i \rvert + \lvert h_{i+1} \rvert + \lvert h_{i+2} \rvert) < 0.14 * \lvert e_j \rvert$	3-To-1
Default	($\lvert e_j \rvert = \lvert h_i \rvert$) $\Vert$ (Rule H1 and E1 Fails)	1-To-1

Examining the data, we observe that it is possible to align one English sentence with one or more Hindi sentences or vice-versa. In the method described below, sentence *length* is calculated in *number of words*. We define our task as that of learning rules that characterise the relationship between the lengths of two sentences in parallel texts. We used 60 manually aligned paragraphs from the EMILLE corpus, each with an average of 3 sentences, as a dataset for our learning task. Initially we derived minimum and maximum length differences in percentages for each of the one-to-one, one-to-two and one-to-three parallel sentence pairs. Later we used these values as input to our algorithm to learn new rules that maximize the probability of aligning sentences.

Learning: Let T = [1:1, 1:2, 1:3, 2:1, 3:1], a set of possible alignment types between the English and Hindi sentences. For each alignment type $t \in T$, minimum and maximum length differences in number of words, normalized to percentages, can be described as $\min_t$ and $\max_t$. For each alignment type $t \in T$, a constant parameter δ_t, where $\delta_t \in [\min_t , \min_t + 0.01, \min_t + 0.02, \ldots, \max_t]$ was learned using an algorithm described in figure 2.1. δ_t is a value that describes the length relationship between the sentences of a pair of type *t*. For example, given a pair of one Hindi and two English sentences and a value δ_t, where t = 1:2, it is possible to check if these sentences can be aligned with each other. Suppose for a given pair of parallel sentences that consist of h_i (Hindi sentence at i^{th} position) and e_j and e_{j+1} (English sentences at j^{th} and $j+1^{th}$ positions), let $|h_i|$, $|e_j|$ and $|e_{j+1}|$ be the lengths of Hindi and English sentences. h_i, e_j and e_{j+1} are said to have 1:2 alignment if $\mathbf{|h_i| - (|e_j| + |e_{j+1}|) < 0.17 * |h_i|}$, i.e. the difference between the length of the Hindi sentence and the length of the two consecutive English sentences is less than ($\delta_{t=1:2} = 0.17$) times the length of the Hindi sentence. Table 2.1 lists rules for different possible alignments. Before we decide on the final alignment, we check each possibility of one Hindi sentence being aligned with one, two or three consecutive English sentences and vice-versa. We use rules H1 and H2 to check the possibility of one Hindi sentence being aligned with two or three consecutive English sentences. Similarly, rules E1 and E2 are used to check the possibility of one English sentence being aligned with two or three consecutive Hindi sentences. If none of the rules from H1, H2, E1 and E2 return true, we consider the default alignment (1-To-1) between the English and Hindi sentences. We give preference to the higher alignment over the possible lower alignments, i.e. given 1-To-2 and 1-To-3 possible alignment mappings, we consider 1-To-3 mapping. We tested our algorithm on parallel texts with total of 3441 English-Hindi sentence pairs and obtained an accuracy of 99.09%; i.e., the correctly aligned pairs were 3410.

3 Word Alignment

Extending sentence alignment to word alignment is a process of locating corresponding word pairs in two languages. In some cases, a word is not translated, or is translated by several words. A word can also be a part of an expression that is translated as a whole, and therefore the entire expression must be translated as a whole (Manning & Schütze, 2003). We present a hybrid method for many-to-many word alignment. Hindi is a partial free order language where the order of word groups in a Hindi sentence is not fixed, but the order of words within groups is fixed (Ray et al., 2003). According to Ray et al. (2003), fixed order word group extraction is essential for decreasing the load on the free word order parser. The word alignment algorithm takes as input a pair of aligned sentences and groups words in sentences of both languages. We have observed a few facts about the Hindi language. For example, there are no articles in Hindi (Bal Anand, 2001). Since there are no articles in Hindi, articles are aligned to null.

3.1 Local word grouping

A separate group is created for each token in the English text. Every English word has one property associated with it: the lemma of the word. This is necessary because a dictionary lookup approach is at the heart of our word alignment algorithm. Verbs are used in different inflected forms in different sentences. For a verb, it is common not to find all inflected forms listed in a dictionary, i.e. most dictionaries contain verbs only in their base forms. Therefore we use a morphological analyzer to find the lemma of each English word.

Word groups in Hindi are created using two resources: a Hindi gazetteer list that contains a large set of named entities (NE) and a rule file that contains more than 250 rules. The gazetteer list is available as a part of Hindi Gazetteer Processing Resource in GATE (Maynard et al., 2003). For each rule in the rule file, it contains the following information:

1. Hindi Regular Expression (RE) for a word or phrase. This must match one or more words in the Hindi sentence.
2. Group name or a part-of-speech category.
3. Expected English word(s) (EEW) that this Hindi word group may align to.
4. Expected Number of English words (NW) that the Hindi group may align to.
5. In case a group of one or more English words aligns with a group of one or more Hindi words, information about the key words (KW) in both groups. Key words must match each other in order to align English-Hindi groups.
6. A rule to convert the Hindi word into its base form (BF).

Rules in the rule file identify verbs, postpositions, noun phrases and also a set of words, whose translation is expected to occur in the same order as the English words in the English sentence. The local word grouping algorithm considers one rule at a time and tries to match the regular expression in the Hindi sentence. If the expression is matched, a separate group for each found pattern is created. When a Hindi group is created, based on its pattern type, one of the following categories is assigned to that group:

proper-noun	city	job-title	location
country	number	day-unit	date-unit
month-unit	verb	auxiliary	pronoun
post-position	other		

These rules have been obtained mainly through consulting Hindi grammar material (Bal Anand, 2001 and Ta, 2002) and by observing the EMILLE corpus. For example, consider the following rules:

No	RE	Cat	EEW	NW	KW	BF
1	बावन	num	fifty two	2		
2	(.)+ रहा	verb			1	
3	(.)+ ते थे	verb			1	1,ते = ना
4	(.)+ के लिये	prep	for (.)+	2	1-2	
5	अलग अलग	other	different	1		

i) "रहा ", "रहे ,""रही" are used to indicate the progressive tense. They can be seen as analogous to the English (-ing) ending.
ii) "ते ,""ता", and "ती" are used as verb endings to indicate the habitual tense. They must agree with subject number and gender.
iii) "थे "is a past tense conjunction of the verb "होना".

In the first rule, if we find a word "बावन" (bavan) in Hindi, we mark it as a "Number" and search for the English string with two words that is equal to the expected string "fifty two". In the second rule, we locate a string where the second word is "रहा" (raha). "1" in the fifth column specifies that the first word is the keyword. We use the dictionary to locate the word in the English sentence that matches with the key word. If the English word is located, we align "(.)+ रहा" with the English word found. In the third rule, if we find a Hindi string with two words where the first word ends with "ते " (te) and the second word is "थे "(the), we group them as a verb. As specified in the sixth column, we replace the characters "ते "with "ना" (na) to convert the first word into its base form (e.g. "गाते "(gaate) into "गाना" (gaana)). In the fourth rule, we align "X के लिये" with "For X", where "For" = "के लिये". As specified in the fifth column, we align the first word in Hindi with the second word in English. In the final example, we group two words that are identical to each other. For example: "अलग अलग" (alag alag) which means "different" in English. Such bigrams are used to stress the importance of a word/activity in a sentence.

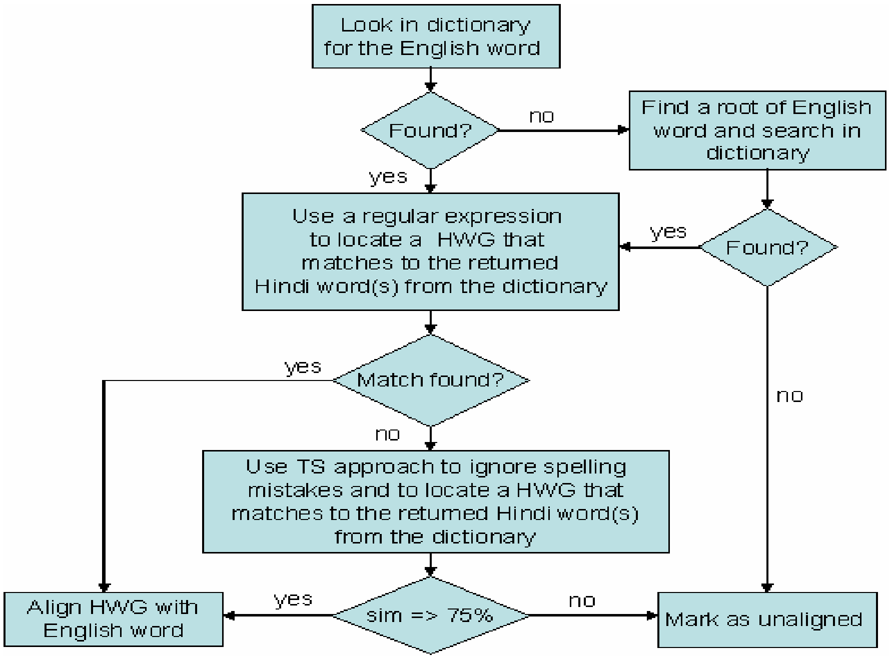

Look in dictionary for the English word
Found?
no
Find a root of English word and search in dictionary
yes
Use a regular expression to locate a HWG that matches to the returned Hindi word(s) from the dictionary
yes
Found?
yes
Match found?
no
no
Use TS approach to ignore spelling mistakes and to locate a HWG that matches to the returned Hindi word(s) from the dictionary
Align HWG with English word
yes
sim => 75%
no
Mark as unaligned

Figure 3.1 Dictionary Lookup Approach

example, in rule 3 and 4 if the word ends with either of ता, ते or ती followed by (PH), it is assumed that the word is a verb. The formula for finding the lemma of any Hindi verb is: **infinitive = root verb + "ना"**. Sometimes it is possible to predict the corresponding English translation. For example, for the postposition "के सामने", one is likely to find the preposition "in front of" in the English sentence. We store this information as an expected English word(s) in Hindi Word Groups (HWGs) and search for it in the English sentence. In the case of rules 4 and 5, though the HWG contains more than one word, only one is the actual verb (key word) that is expected to be available in a dictionary. We specify the index of this key word in the HWG, so as to consider only the word at the specified index to compare with key word in English word group. If they match, the full HWG is aligned to the word in English sentence.

3.2 Alignment Algorithm

After applying the local word grouping rules to the Hindi sentence(s), based on their categories of HWGs, we use four methods to process and align HWGs with their respective English Word Groups.

1. Dictionary lookup approach (DL)
2. Transliteration similarity approach (TS)
3. Expected English words approach (EEW)
4. Nearest aligned neighbour approach

Whilst the verbs and other groups are processed with DL approach, HWGs with categories such as proper nouns, city, job-title, location, and country are processed with TS approach. HWGs such as number, day-unit, date-unit, month-unit, auxiliary, pronoun and postpositions, where the expected English words are specified, are processed with EEW approach. Sometimes the combination of DL and TS is also used to identify the proper alignment. At the end, nearest aligned neighbour approach is used to align the unaligned HWGs.

Dictionary Lookup

The corpus we used in our experiments is encoded in Unicode and therefore the word matching process requires dictionary entries to be in Unicode encoding. The only English-Hindi dictionary we found is called, "**shabdakoSha"** and is freely available from (WWW2). In this dictionary, the ITRANS transliteration system is followed, i.e. Hindi entries are not written in the Devanagari script, but in the Roman script. This dictionary has around 15,000 English words, each with an average of 4 relevant Hindi words. Following

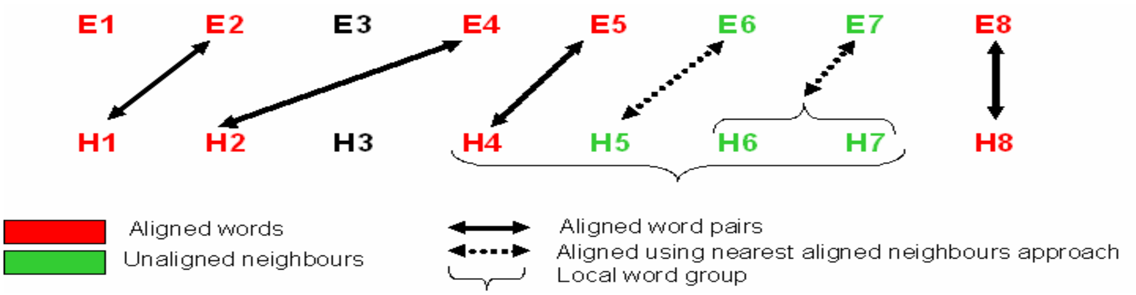


Figure 3.2 Nearest Aligned Neighbours Approach

ITRANS conventions, a parser was developed to convert all these entries into Unicode. Given a set of English and Hindi words, the algorithm presented in figure 3.1 is executed to search for the best translation among the English words.

Transliteration Similarity

A transliteration system maintains a consistent correspondence between the alphabets of two languages, irrespective of sound (Manning & Schütze, 2003). Given two words, each from a different language, we define "transliteration similarity" as the measure of likeness between them. This could exist due to the word in one language being inherited or adopted by the other language, or because the word is a proper noun. Named entities such as city, job-title, location, country and proper nouns, all recognized by the local word grouping algorithm are compared using a transliteration similarity approach. This likeness is counted using a table that lists letter correspondences between the alphabets of two languages. For the English and Hindi languages, it is possible to come up with a table that defines letter correspondence between the alphabets of two languages. For example,

A → अ, B → ब, Bh → भ, Ch → च,

D → द, Dh → ध and so on…

A bidirectional mapping is established between each character in the English and Hindi alphabets. When DL is not able to find any specific English word in dictionary, this approach is used to find the transliteration similarity between the unaligned words. Sometimes because the words in a Hindi sentence are not spelled correctly, when DL issues a query to dictionary, none of the Hindi words appearing in a Hindi sentence match with the words returned from dictionary. We use a dynamic programming algorithm "edit-distance" to calculate similarity between these words (WWW3). According to WWW3, *"The edit distance of two strings, s1 and s2, is defined as the* ***minimum*** *number of* ***point mutations*** *required to change s1 into s2, where a point mutation is one of: change a letter, insert a letter or delete a letter."* The lower the distance, the greater the similarity. From our experiments of 100 proper noun pairs, we found that if the similarity is greater than 75%, the words can be reliably aligned with each other. We consider a pair with the highest similarity. E.g.: **Aswani → अासवानी.** Here we remove vowels in both strings, except those that appear at the start of words. After the removal of vowels from the English and Hindi texts, the resulting text would be: **Aswn → असवन.** The Hindi text is then converted into English text using the transliteration table: **Aswn → Aswn.** The two texts are then compared using an "edit-distance" algorithm.

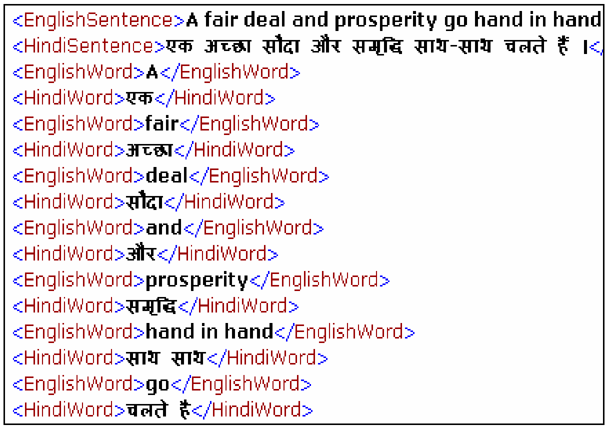
Expected English word(s)

For HWGs which are categorised as numbers, job-titles or postpositions, it is possible to specify the expected English word or words that can be found in the parallel English text. The algorithm retrieves expected English word(s) from the HWGs and tries to locate them in the English sentence. This approach can be useful to locate one or more English words that align with one or more Hindi words. For example, the number "बयालिस" whose equivalent translation in English is "forty two" has two words in English, and the postposition "के सामने", whose equivalent translation in English is "in front of", has three words in English. These are examples of many-to-many word alignment.

Nearest Aligned Neighbours

At the end of the first three stages of the word alignment process, many words remain unaligned. Here we introduce a new approach, called the "Nearest Aligned Neighbours approach". In certain cases, words in English-Hindi phrases follow a similar order. The Nearest Aligned Neighbours approach works on this principle and aligns one or more words with one of the English words. A local word grouping algorithm, explained in section 3.1, groups such phrases and tags them as "group". Considering one HWG at a time, we find the nearest Hindi word that is already aligned with one or more English word(s). We assume that the words in English-Hindi phrases follow a similar order and align the rest words in that group accordingly. An example of alignment using the Nearest Aligned Neighbours approach is given in Figure 3.2. Word H4 is already aligned with E5, and H3, H5, H6 and H7 are yet to be aligned. The local word grouping algorithm has tagged a sequence of H4, H5, H6 and H7 as a single group. At the same time, H6 and H7 are also grouped as a single group. The algorithm searches for the aligned Hindi word, which, in this case, is H4 and aligns H5 with E6 and the group of H6 and H7 with E7.

4 Results



```
<EnglishSentence>A fair deal and prosperity go hand in hand
<HindiSentence>एक अच्छा सौदा और समृद्धि साथ-साथ चलते हैं ।<
<EnglishWord>A</EnglishWord>
<HindiWord>एक</HindiWord>
<EnglishWord>fair</EnglishWord>
<HindiWord>अच्छा</HindiWord>
<EnglishWord>deal</EnglishWord>
<HindiWord>सौदा</HindiWord>
<EnglishWord>and</EnglishWord>
<HindiWord>और</HindiWord>
<EnglishWord>prosperity</EnglishWord>
<HindiWord>समृद्धि</HindiWord>
<EnglishWord>hand in hand</EnglishWord>
<HindiWord>साथ साथ</HindiWord>
<EnglishWord>go</EnglishWord>
<HindiWord>चलते हैं</HindiWord>
```

Figure 4.1 Word Alignment Results

We performed manual evaluation of our word alignment algorithm on a set of parallel data aligned at the sentence level. The parallel texts consist of 3954 English and 5361 Hindi words taken from the EMILLE Corpus. We calculate our results in terms of the number of aligned English word groups. The precision is calculated as the ratio of the number of correctly aligned English word groups to the total number of English word groups aligned by the system, and recall is calculated as the ratio of the number of correctly aligned English word groups to the total number of English word groups created by the system. We obtained 77% precision and 67.79% recall for many-to-many word alignment. Figure 4.1 shows an example of the word alignment results.

5 Future works

It would be useful to evaluate separate stages (i.e. DL, TS, EEW and Nearest Aligned Neighbours approach) in the word alignment algorithm separately. We aim to do this as part of a failure analysis of the algorithm in future. We also aim to improve our alignment results by using Part-of-Speech information for the English texts. We aim to implement or use local word grouping rules for the English text and improve our existing word grouping rules for the Hindi texts. The Nearest Aligned Neighbours approach suggests possible alignments, but we are trying to integrate some statistical ranking algorithms in order to suggest more reliable pairs of alignment. Yarowsky et al. (2001) introduced a new method for developing a Part-of-Speech tagger by projecting tags across aligned corpora. They used this technique to supply data for a supervised learning technique to acquire a French part-of-speech tagger. We aim to use our English-Hindi word alignment results to bootstrap a Part-of-Speech tagger for the Hindi language.

References

Bal Anand, 2001, *Hindi Grammar Books for standard 5 to standard 10*, Navneet Press, India.

Baker P., Bontcheva K., Cunningham H., Gaizauskas R., Hamza O., Hardie A., Jayaram B.D., Leisher M., McEnery A.M., Maynard D., Tablan V., Ursu C., Xiao Z., 2004, *Corpus linguistics and South Asian languages: Corpus creation and tool development,* Literary and Linguistic Computing, 19(4), pp. 509-524.

Brown, P., Lai, J. C., and Mercer, R., 1991, *Aligning Sentences in Parallel Corpora*, In Proceedings of ACL-91, Berkeley CA.

Chen S., 1993, *Aligning sentences in bilingual corpora using lexical information*, Proceedings of the 31st conference on Association for Computational Linguistics**,** pp. 9 – 16, Columbus, Ohio.

Gale W., and Church K., 1993, *A program for aligning sentences in bilingual corpora,* Proceedings of the 29th conference of the Association for Computational Linguistics, pp.177-184, June 18-21, 1991, Berkeley, California.

Haruno M. and Yamazaki T., 1996, *High-performance bilingual text alignment using statistical and dictionary information*, Proceedings of the 34th conference of the Association for Computational Linguistics, pp. 131 – 138, Santa Cruz, California.

Kay M. and Roscheisen M., 1993, *Text translation alignment*, Computational Linguistics, 19(1):75--102.

Manning C. and Schütze H., 2003, *Foundations of Statistical Natural Language Processing,* MIT Press, Cambridge, Massachusetts.

Mark D., 2004, *Technical Report on Unicode Standard Annex #29 - Text Boundaries*, Version 4.0.1, Unicode Inc., http://www.unicode.org/reports/tr29/ [22/11/04].

Mayers A., Grishman R., Kosaka M., 1998, *A Multilingual Procedure for Dictionary-Based Sentence Alignment*, Proceedings of the Third Conference of the Association for Machine Translation in the Americas on Machine Translation and the Information Soup.

Maynard D., Tablan V., Bontcheva K., Cunningham H., 2003, *Rapid customisation of an Information Extraction system for surprise languages,* ACM Transactions on Asian Language Information Processing, Special issue on Rapid Development of Language Capabilities: The Surprise Languages.

Ray, P, Harish V., Sarkar, S., and Basu, A., 2003, *Part of Speech Tagging and Local Word Grouping Techniques for Natural Language Parsing in Hindi*, Proceedings of the 1st International Conference on Natural Language Processing (ICON 2003); Mysore.

Simard M. and Pierre P., 1996, *Bilingual Sentence Alignment: Balancing Robustness and Accuracy*, Proceedings of the First Conference of the Association for Machine Translation in the Americas (AMTA-96), pp. 135-144, Montreal, Quebec, Canada.

Ta A., 2002, *A Door into Hindi*, NC State University, http://www.ncsu.edu/project/hindi_lessons/lessons.html [22/11/04]

Warwick S., Catizone, R., and Graham R., 1989, *Deriving Translation Data from Bilingual Texts*, in Proceedings of the First International Lexical Acquisition Workshop, Detroit.

WU D., Jul 2000, *Alignment,* In Robert DALE, Hermann MOISL, and Harold SOMERS (editors), Handbook of Natural Language Processing. pp. 415-458. New York: Marcel Dekker. ISBN 0-8247-9000-6.

WWW1, *Devanagari Unicode Chart, the Unicode Standard*, Version 4.0, Unicode Inc.,http://www.unicode.org/charts/PDF/U0900.pdf [22/03/05].

WWW2, *English-Hindi dictionary source*, http://sanskrit.gde.to/hindi/dict/eng-hin_guj.itx [22/03/05].

WWW3, *Dynamic Programming Algorithm (DPA) for Edit-Distance*, http://www.csse.monash.edu.au/~lloyd/tildeAlgDS/Dynamic/Edit/ [22/03/05]

Yarowsky, D., G. Ngai and R. Wicentowski, 2001, *Inducing Multilingual Text Analysis Tools via Robust Projection across Aligned Corpora,* In Proceedings of HLT 2001, First International Conference on Human Language Technology Research.