Using Movie Subtitles for Creating a Large-Scale Bilingual Corpora

Einav Itamar, Alon Itai

Computer Science Department
Technion – Israel Institute of Technology
32000 Haifa, Israel
itamar@cs.technion.ac.il, itai@cs.technion.ac.il

Abstract

This paper presents a method for compiling a large-scale bilingual corpus from a database of movie subtitles. To create the corpus, we propose an algorithm based on Gale and Church's sentence alignment algorithm(1993). However, our algorithm not only relies on character length information, but also uses subtitle-timing information, which is encoded in the subtitle files. Timing is highly correlated between subtitles in different versions (for the same movie), since subtitles that match should be displayed at the same time. However, the absolute time values can't be used for alignment, since the timing is usually specified by frame numbers and not by real time, and converting it to real time values is not always possible, hence we use normalized subtitle duration instead. This results in a significant reduction in the alignment error rate.

1. Introduction

In our age, where vast electronic corpora deposits abound, the real challenge is to harvest resources to create useful tools. In almost any statistical machine translation system, there is a need for large aligned bilingual corpora.

To acquire a large bilingual corpus we mine a resource which until recently has not been utilized for NLP tasks – movie subtitles. It is then shown how information specific to this media can be used to automatically align the corpora.

The main advantages of using subtitles are:

1. They can be obtained in an almost infinite amount (which grows daily).
2. They are publicly available and can be downloaded freely from a variety of subtitle web sites.
3. They are available in many languages.
4. The subtitle files contain timing information which can be exploited to significantly improve the quality of the alignment.
5. Translated subtitles are very similar to those in the original language – contrary to many other textual resources, the translator must adhere to the transcript and can't skip, rewrite, or reorder paragraphs.

Therefore, building large-scale bilingual corpora using movie subtitle files is pretty straightforward: Subtitle translations of many popular American movies are available in many languages, even rare ones, thus creating a parallel corpus for, let's say, Slovak and Greek can be easily done by collecting translated subtitles for those languages, and using our method for building an aligned corpus from the subtitle files.

First, we will discuss the previous work that was done in this area. Then, we describe the data that we used, and its special characteristics. In Section 4. we present our method to align the subtitle files by using timing information.

2. Previous Work

Mangeot and Giguet (2005) were the first to present a methodology for building aligned multilingual corpora from movie subtitles. The subtitles were downloaded from the web and were aligned with the time used to display them on the screen. The whole process was semi-automatic – a human operator had to synchronize some of the subtitles. Lavecchia et al. (2007) constructed an English-French parallel corpus using a method they refer to as Dynamic Time Warping. The corpus was quite small, consisting only 37,625 aligned subtitle pairs which were collected from 40 movies. Tiedemann (2007) created a multilingual parallel corpus of movie subtitles using roughly 23,000 pairs of aligned subtitles covering about 2,700 movies in 29 languages. Tiedemann proposed an alignment approach based on time overlaps. In order to overcome timing encoding problems (see Section 3.), he uses human intervention in order to set the speed and the offset of the subtitles and also to fix time shifts which are usually caused by frame rate conversions. In contrast our alignment is fully automatic, no human intervention is necessary and thus we can align subtitles for languages we do not know.

3. The Data

3.1. Technical Details

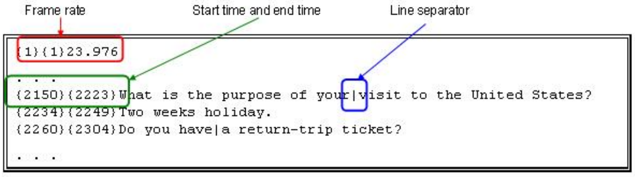
About 100,000 subtitle files were downloaded from http://www.opensubtitles.org[1]. Each file contains a translation of a movie to some language, and after sorting (first by movie name then by language) we got subtitle files for 19,715 movies. Table 1 describes the number of distinct movies per language.

The files are textual, and contain a list of subtitles. Each subtitle is composed of 1-3 lines and timing information, which is required for rendering the subtitles onto the movie itself (see Figure 1). Some of the subtitles are split over two or more files.

[1] Thanks to Branislav Gerzo, the administrator of the site, for his support.

Language	Number of Movies
English	12044
Spanish	7065
French	5738
Czech	5216
Dutch	5152
Serbian	3752
Croatian	3619
Portuguese	3325
Brazilian P.	3288
Polish	3220
Slovenian	2996
Turkish	2800
Romanian	2740
Swedish	2707
Bulgarian	2696
Danish	2373
Finnish	2286
Greek	2205
Hungarian	1862
German	1761
Estonian	1422
Arabic	1405
Hebrew	1374
Italian	1279
Norwegian	1182
Slovak	1104
Russian	1048
Macedonian	442
Chinese	325
Korean	299
Icelandic	241
Bosnian	204
Albanian	198
Lithuanian	150
Japanese	78

Table 1: Number of movies in our monolingual corpora



```
{1}{1}23.976

. . .
{2150}{2223}What is the purpose of your|visit to the United States?
{2234}{2249}Two weeks holiday.
{2260}{2304}Do you have|a return-trip ticket?

. . .
```

Figure 1: Subtitle File Format

Each subtitle s has a start time $start(\mathrm{s})$ and an end time $end(\mathrm{s})$. Some of the formats encode these values as real time, and some as frame numbers. In order to convert frame numbers to absolute time, the frame rate of the movie file itself is required. However, not all subtitle files contain the correct frame rate, and some omit it altogether.

Most of the subtitle files contain many OCR (Optical Character Recognition) errors, probably because some of them were scanned from the movie DVD as an image. Another technical difficulty is that sometimes there are different editions for the same movie (Unrated version, Director's cut, etc.) or even worse – sometimes two different movies have the same name.

	Subtitles		BNC		Conversational	
Token	Freq.	Rank	Freq.	Rank	Freq.	Rank
you	33080	1	6674	15	32085	2
i	30029	2	8513	13	39817	1
the	23092	3	60405	1	27351	4
to	16839	4	24957	4	16928	8
it	14831	5	10527	8	30417	3
that	10900	6	10805	7	19964	6
n't	10390	7	3165	38	18418	7
and	9791	8	26123	3	21569	5
do	9598	9	2700	41	12854	10
of	8595	10	28830	2	8332	17
what	8052	11	2398	48	9138	15
is	7817	12	9855	9	8343	16
in	7576	13	18406	6	10107	13
me	7413	14	1287	78	3898	48
we	7208	15	3498	34	7869	21
this	6731	16	4532	23	4734	33
he	5826	17	6400	17	11465	11
on	5824	18	7049	19	7951	20
my	5448	19	1466	72	2857	58
your	5289	20	1342	74	3435	53

Table 2: Rank and frequency counts for the 20 most common words in our corpus and the frequencies of the corresponding words in the BNC and the BNC conversational corpora (frequencies are per million words).

3.2. The Genre

The genre of the text is quite unique. Since the scripts were "meant to be spoken", they resemble spoken more than written language: many sentences are incomplete, and personal pronouns abound. On the other hand, it also differs from spoken language, since the subtitles were composed by professional script writers, whereas spoken language corpora consist of transcript of spontaneous conversations between native speakers and usually contain many false starts, repeats, breaks, etc. We compared the frequencies of the 20 most frequent words in our corpus to the frequency counts of the BNC (British National Corpus) (Leech, G. et al., 2001) and to the counts of the conversational part of the BNC.

As can be seen in Table 2, the frequency of pronouns in our corpus is significantly higher than the corresponding frequencies in the BNC. The reason for this is that pronouns are used extensively in dialogues. The word "you", for example, which is the top ranked word in our corpus, covers 3.3% of the corpus. In the BNC, however, it is ranked only 15^{th}, and its frequency is only 0.66%. The frequency counts of the BNC conversational English corpora, are much more similar to ours. However, for some words the counts differ significantly, thus showing that the genre of the language of movies is different than conversational language,

	Easy R	document P	Noisy R	document P
Our cost function	0.98	0.97	0.78	0.74
Gale and Church's cost function	0.99	0.97	0.65	0.60

Table 3: Comparison of our cost function to Gale and Church's in terms of recall(R) and precision(P)

3.3. Building the Corpus

In order to build the corpus we need to select two languages for building a subtitle aligned bilingual corpus. For each language a single version of the movie translation should be selected (for some of the movies we found more than one version). The pair of versions should be as similar as possible – the more similar they are the easier it is to align them and to get better statistics. For that, all possible pairs of versions are aligned and the pair that shows the minimal alignment cost is selected.

4. Aligning the Corpus

We wish to find which subtitles in the first language correspond to subtitles in the second language. Suppose we want to align two subtitle files for the same movie, each in a different language. Let $S_e = (\mathbf{e}^1 \dots \mathbf{e}^{|S_e|})$ and $S_f = (\mathbf{f}^1 \dots \mathbf{f}^{|S_f|})$ be the subtitles of the first and second languages respectively. We wish to find which subtitles are translations of one another.

4.1. The Dynamic Programming Algorithm

The task of subtitle alignment is similar to sentence alignment – most of the translations are one-to-one, there are some deletions (0:1 or 1:0) and some one-to-many (1:2, 2:1, etc.). Since many-to-many translations are quite rare, for performance reasons, we decided to allow only 1:1, 0:1, 1:0, 2:1 and 1:2 alignments, and (naturally) restricted the alignment to contain only non-crossing sentence pairs. We used dynamic programming to find a minimal cost alignment satisfying these constraints. The recursive definition of matching cost is similar to that of (Gale and Church, 1993):

$$C(i,j) = min \begin{cases} C(i-1,j-1) & + \quad c(\mathbf{e}^i, \mathbf{f}^j) \\ C(i-1,j) & + \quad c(\mathbf{e}^i, \phi) \\ C(i,j-1) & + \quad c(\phi, \mathbf{f}^j) \\ C(i-2,j-1) & + \quad c(\mathbf{e}^{i-1}||\mathbf{e}^i, \mathbf{f}^j) \\ C(i-1,j-2) & + \quad c(\mathbf{e}^i, \mathbf{f}^{j-1}||\mathbf{f}^j)\ , \end{cases}$$

where $C(i,j)$ is the cost of aligning $\mathbf{e}^1 \dots \mathbf{e}^i$ with $\mathbf{f}^1 \dots \mathbf{f}^j$, $c(\mathbf{e}, \mathbf{f})$ is the cost of aligning $\mathbf{e}$ with $\mathbf{f}$, ϕ is an empty string and $\mathbf{x}||\mathbf{y}$ is the concatenation of $\mathbf{x}$ with $\mathbf{y}$.

4.2. The Cost Function

In their work, Gale and Church defined $c(\mathbf{e}, \mathbf{f})$ by means of relative normalized length of sentence in characters, namely $\frac{l(\mathbf{e})}{l(S_e)}$ and $\frac{l(\mathbf{f})}{l(S_f)}$ where $l(S_e)$ and $l(S_f)$ are the total lengths of the subtitle files of the first and second languages, respectively.

Following Gale and Church, we used length in characters as a metric for similarity, but we also used an additional resource which is not available in 'traditional' corpora – *timing information*. Timing is highly correlated between subtitles in different versions (for the same movie), since subtitles that match should be displayed at the same time. However, the absolute time values can't be used for alignment, since the timing is usually specified by frame numbers and not by real time, and converting it to real time values is not always possible. Therefore, we define the duration $d(\mathbf{s})$ of subtitle $\mathbf{s}$:

$$d(\mathbf{s}) = end(\mathbf{s}) - begin(\mathbf{s})\ .$$

Again, the data should be normalized to the average display time, and the cost should be measured in absolute values, therefore we have chosen:

$$c(\mathbf{e}, \mathbf{f}) = \lambda \left(\frac{d(\mathbf{e})}{d(S_e)} - \frac{d(\mathbf{f})}{d(S_f)} \right)^2 + (1-\lambda) \left(\frac{l(\mathbf{e})}{l(S_e)} - \frac{l(\mathbf{f})}{l(S_f)} \right)^2$$

as our cost function, where $d(S_e)$ and $d(S_f)$ are the total duration of the first and the second language versions, and $0 \leq \lambda \leq 1$ is a parameter that represents the relative importance of the timing information. Assuming that the probability that a subtitle will appear in only one of the versions does not depend on character length or subtitle duration, we assign:

$$c(\mathbf{e}, \phi) = c(\phi, \mathbf{f}) = \delta$$

for all zero-to-one alignments.

Both δ and λ are language dependent parameters whose optimal values were determined empirically by a grid search. For English-Hebrew their values were $\lambda = 0.63$ and $\delta = 0.4$.

The same procedure was performed on another pair of languages – English-Spanish. This time we used only a "noisy" document for estimating the parameters. The optimal λ of this experiment was similar to that of the previous experiment, thus supporting the hypothesis that this parameter is not language-pair dependant. This is not surprising, since the relative weight of the timing information is not expected to be language dependent. The optimal δ, however, was slightly higher (0.47) which might indicate that for more similar languages it is better to have a higher penalty for skipping.

5. Experiments

In order to verify that timing information is useful, we compared the performance of our cost function to that of Gale and Church. Mathematically, an *alignment* is a set of pairs of aligned subtitles from the two languages. Let A be the set of the alignments generated by the algorithm and B is the set of the manually determined correct alignments. We calculated the performance in terms of both recall R and precision P, defined as:

$$R = \frac{|A \cap B|}{|B|}\ , \quad P = \frac{|A \cap B|}{|A|}\ .$$

The experiment was conducted on English-Hebrew versions. We compared the performance on both an "easily alignable" document, which was mainly composed of 1:1 alignments, and on a "noisy" document, which contained many 1:0, 2:1 and 2:2 alignments. The first contained 573 alignments and the second contained 737 alignments. As can be seen in Table 3, on the easy document the performance was roughly equal. However, on the noisy document our cost function performed much better. The statistical significance of the difference is more than 99% (for both recall and precision). This indicates (see also (Church, 1993)) that when the input is noisy, length based methods tend to break down. Hence, timing information is essential when the alignment task is not trivial.

6. Conclusions

This paper shows how to harvest a large bilingual resource – movie subtitles, and how to utilize the timing information to obtain a good quality alignment. In a companion paper, we extend the subtitle alignment to the word/phrase level. The result of the latter alignment can be used to further improve the results of the subtitle alignment – two subtitles whose words don't match should probably not be aligned. Thus the alignment of the subtitles should be changed to improve the original subtitle alignment.

7. References

Kenneth Ward Church. 1993. Char_align: a program for aligning parallel texts at the character level. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Comput. Linguist.*, 19(1):75–102.

Caroline Lavecchia, Kamel Smaïli, and David Langlois. 2007. Building parallel corpora from movies. In *Proceedings of The 5th International Workshop on Natural Language Processing and Cognitive Science*.

Leech, G., Rayson, P., and Wilson, A. 2001. *Word Frequencies in Written and Spoken English: Based on the British National Corpus*. Pearson ESL.

Mathieu Mangeot and Emmanuel Giguet. 2005. Multilingual aligned corpora from movie subtitles. Technical report, Condillac-LISTIC.

Jörg Tiedemann. 2007. Building a multilingual parallel subtitle corpus. In *Proceedings of CLIN 17*.