Chapter 2
Implementation

This chapter introduces the general design characteristics of PRESEMT and provides a detailed description of all resources required as well as all pre-processing steps needed, such as corpora processing and model creation.

2.1 Introduction: Summary of the Approach

PRESEMT proposes a novel paradigm, which supports the straightforward development of MT systems for new language pairs using only a limited size of linguistic resources, by applying pattern recognition principles in a modular architecture. With respect to corpora, PRESEMT relies on two different sources of linguistic content, a large TL monolingual corpus and a small parallel corpus, typically comprising of a few hundred aligned SL-TL sentence pairs. Such resources can easily be collected from the Web, because one can easily find plenty of monolingual corpora for almost any language and the parallel corpus required is so small that it can even be assembled by hand. Therefore, PRESEMT overcomes one of the most important bottlenecks of all statistical systems (Munteanu and Marcu 2005): the availability of large parallel corpora of adequate quality. Such corpora are hard to find, particularly when not so widely used languages, such as Greek and Norwegian, are involved. The quality of the translation of such systems depends on the size and quality of the parallel corpus. Even if such corpora exist, they are frequently restricted to a very specific domain, such as the European Parliament (Koehn 2005). In addition to the size, the quality of the parallel corpus is also very important, but this is a factor that most MT research papers avoid to touch, as statistical methods promise that data size overcomes any minor quality issues that the data might have. But this is an important issue for parallel corpora that involve less popular languages, as size is comparably limited and quality-related issues usually increase as the text used is increased. In PRESEMT, both corpora are processed by their respective modules to produce the resources and models required

G. Tambouratzis et al., *Machine Translation with Minimal Reliance on Parallel Resources*, SpringerBriefs in Statistics,
DOI 10.1007/978-3-319-63107-3_2

by the translation process. Finally, the modular architecture allows the language pair developer to replace one algorithm for another, as long as the strict requirements regarding data input and output formats are met.

Besides the two corpora described above, PRESEMT requires a bilingual lemma dictionary, a tagger-lemmatiser in both languages, and a shallow parser in the target language, that divides the text into non-overlapping sub-sentential segments (linguistic chunks). The system uses the TL parser to map this information to SL. In other words, given a parser (or more generally a phrasing model) in TL, one can generate an appropriate phrasing model in SL using pattern recognition-based clustering techniques. This is achieved in PRESEMT by using the parallel corpus to learn structural correspondences between the two languages in order to create sub-sentential segments which correspond to one another based on the structure of the parallel sentence. The modules implementing this functionality are the Phrase Aligner Module (PAM) and the Phrasing Model Generation (PMG).

Using the linguistic information provided by the shallow parser and the PAM + PMG combination, PRESEMT breaks down the translation process in two separate steps, the first one handling the order of both (i) chunks and (ii) out-of-chunk words in the final sentence, as well as the disambiguation of the more important tokens in the sentence, while the second step produces in parallel translations for all the chunks in the sentence. These two steps that breakdown the translation process in a divide-and-conquer fashion are implemented in the Structure Selection (SS) and the Translation Equivalent Selection (TES) modules, respectively, as discussed in Chap. 3. Figure 2.1 provides an overview of the PRESEMT translation process and the way that resources feed the modules. The following section describes all resources required to generate a working MT system via PRESEMT and all pre-processing steps needed.

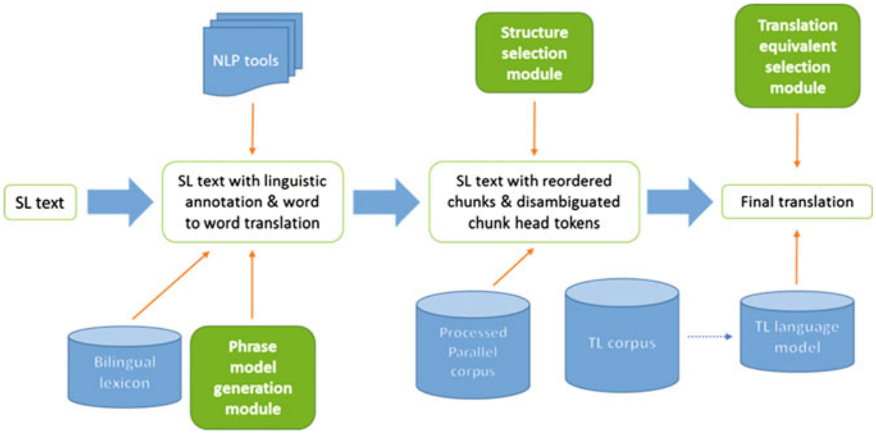

NLP tools
Structure selection module
Translation equivalent selection module
SL text
SL text with linguistic annotation & word to word translation
SL text with reordered chunks & disambiguated chunk head tokens
Final translation
Bilingual lexicon
Phrase model generation module
Processed Parallel corpus
TL corpus
TL language model

Fig. 2.1 Schematic representation of the translation process in PRESEMT

2.2 Linguistic Resources: Data and Existing Linguistic Tools

The system description starts by identifying the required linguistic resources and thus delineating a number of design decisions. A complete list of language-related information is provided, to indicate the basis upon which the specific MT system is created. This list comprises (i) a very small parallel corpus (a few hundred sentence pairs), (ii) a large monolingual corpus of the TL language (where arbitrarily large document collections can be handled) and (iii) a bilingual dictionary with lemmas. Additional linguistic information is provided in the form of Part-of-Speech (PoS) tagging and lemmatising for both languages, and a shallow parser for the TL language with phrase head information. Thus, only minimal resources are required for a prototype translation system to be created. A detailed record of all required linguistic resources follows.

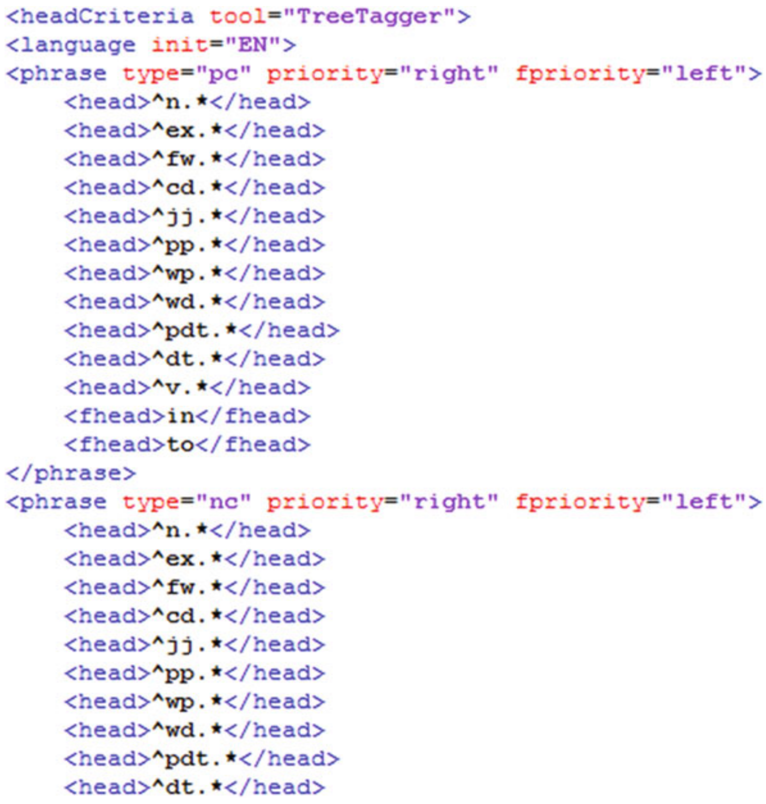
2.2.1 External Processing Tools

The PRESEMT methodology relies on the application of linguistic annotation to all resources: corpora (monolingual and bilingual) and dictionaries, on the basis that resources of smaller size (in relation to those used by traditional SMT systems) may provide more useful information for using in an automatic translation task if linguistically annotated, and not just used to extract n-gram tables. The tools required are a tagger-lemmatiser for both languages and a shallow parser for only the target language.

Lemmatisation and Part-of-Speech tagging helps counterbalance data sparseness, which is a very important issue for PRESEMT because of the limited size of the resources used. Depending on the available tools as well as the language itself, part-of-speech tagging might contain morphological information such as case, number, gender and tense, all of which can greatly improve translation quality.

Syntactic phrases (chunks) give a glimpse of how sentences are formed by providing a flat structure annotation, with the main categories being noun and verb chunks and possibly with clause boundaries. While translating from one language to the other, chunking might also provide information about how groups of words might move from one side of the sentence to the other. Parsing for the source language is derived from the TL parsing scheme using the PMG module, because this way we tackle the problem of having completely different phrase segmentation between source and target languages.

All tools adopted for use in PRESEMT are pre-existing ones instead of tools being designed with the application in mind. Furthermore, these tools are not modified before their integration, so as to perform a realistic evaluation of the worthiness of the PRESEMT concept. This means of course that any errors produced by those tools propagate through the translation process. In order for a tagger-lemmatiser or a chunker to be used in the system, we must provide an interface for the system to interact with the existing tool, that being a simple script, installable software or a Web service. This interface is provided by building a java wrapper using as a guide one of the existing java wrappers created for tools such as the FBT tagger for Greek (Prokopidis et al. 2011) or the TreeTagger (Schmid 1994), which are included in the PRESEMT software package. In the specific case of shallow parsers, along with the java wrapper the developer must also build an accompanying resource for the identification of the head and functioning head (if available) for each phrase type, in the form of an XML file containing regular expressions. This information is essential, as head tokens provide crucial information in the whole translation pipeline. Figure 2.2 provides a sample of the HeadCriteria.xml file.



```
<headCriteria tool="TreeTagger">
<language init="EN">
<phrase type="pc" priority="right" fpriority="left">
    <head>^n.*</head>
    <head>^ex.*</head>
    <head>^fw.*</head>
    <head>^cd.*</head>
    <head>^jj.*</head>
    <head>^pp.*</head>
    <head>^wp.*</head>
    <head>^wd.*</head>
    <head>^pdt.*</head>
    <head>^dt.*</head>
    <head>^v.*</head>
    <fhead>in</fhead>
    <fhead>to</fhead>
</phrase>
<phrase type="nc" priority="right" fpriority="left">
    <head>^n.*</head>
    <head>^ex.*</head>
    <head>^fw.*</head>
    <head>^cd.*</head>
    <head>^jj.*</head>
    <head>^pp.*</head>
    <head>^wp.*</head>
    <head>^wd.*</head>
    <head>^pdt.*</head>
    <head>^dt.*</head>
```

Fig. 2.2 Head criteria file for the TreeTagger in English

2.2.2 Lemma-Based Bilingual Dictionary

The bilingual dictionary contains lemma forms of single-word and multi-word SL-TL lexical correspondences. In addition, it contains linguistic annotations, namely Part-of-Speech tags. The dictionary is a very important resource in the PRESEMT translation methodology and must provide a wide coverage of the source language to support a good translation quality. In theory, the larger the dictionary size, the fewer the out-of-vocabulary (OOV) words are expected to be. In PRESEMT, most of the dictionaries were based on ones provided by publishers and did not contain any linguistic annotations in appropriate and systematic ways, so a pre-processing step was necessary before using them in the MT system.

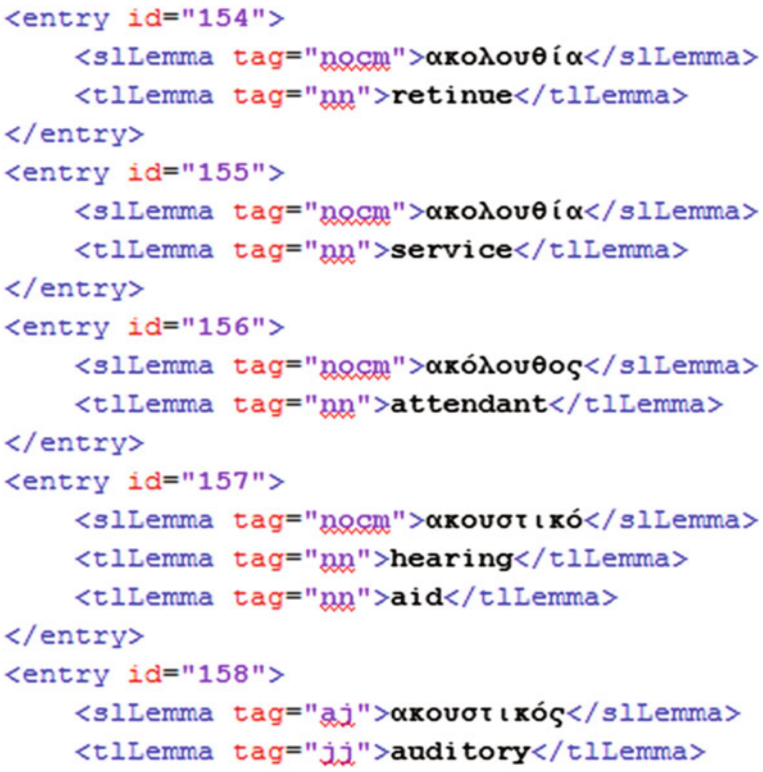
Table 2.1 provides the sizes of the dictionaries used during the PRESEMT project. As can be seen, different dictionary sizes have been adopted, depending on availability. For the system to be able to use a dictionary, this needs to be provided in the respective format. Figure 2.3 shows the XML representation used for storing the dictionaries in PRESEMT.

2.2.3 The Parallel Corpus

The parallel corpus used in PRESEMT needs to contain only a few hundred sentences, as it is only used for mapping the transfer from SL to TL sentence structures, determined as sequences of phrases. The small size of the corpus reduces reliance on costly linguistic resources. The corpus is assembled either from available parallel corpora or by using a Web crawler (Pomikálek and Rychlý 2008) and then manually replacing free translations with more literal ones, to allow the accurate extraction of structural modifications. After building the parallel corpus, we process the source and target side, using the SL and TL tagger-lemmatisers and the TL shallow parser, so as to annotate it with linguistic information. The result is a source and target side incorporating lemma and PoS information and other salient language-specific morphological features (e.g. case, number, tense, etc.) depending on the morphology of the language and the available tools. Furthermore, for the TL

Table 2.1 Dictionaries size for various language pairs used in PRESEMT

Language pair	Source	Number of entries
Greek-English	Developed in other project	40,000
Greek-German	Publisher	80,000
English-German	Developed in other project	1,000,000
Norwegian-English	Publisher	45,000
Norwegian-German	Publisher	37,000
Czech-English	Publisher	180,000
Czech-German	Publisher	70,000

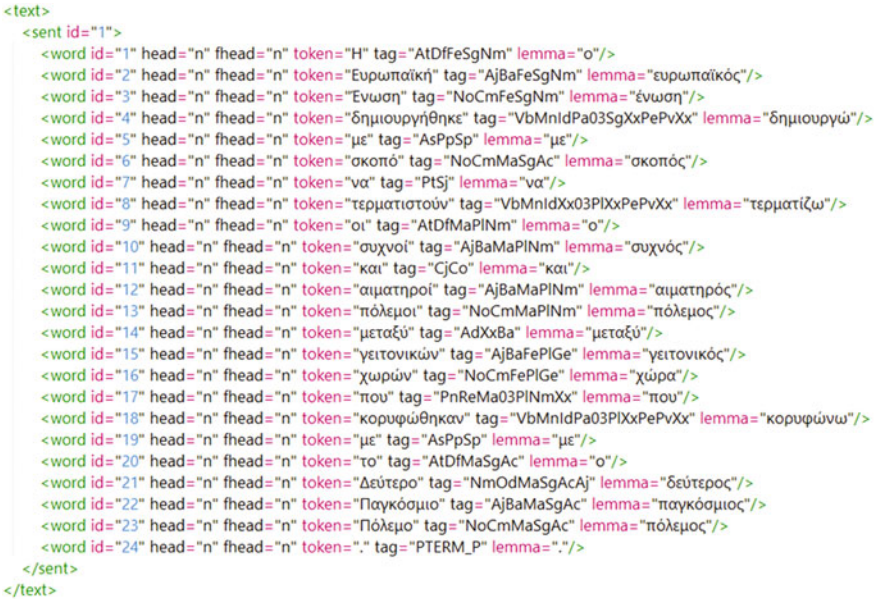


```
<entry id="154">
    <slLemma tag="nocm">ακολουθία</slLemma>
    <tlLemma tag="nn">retinue</tlLemma>
</entry>
<entry id="155">
    <slLemma tag="nocm">ακολουθία</slLemma>
    <tlLemma tag="nn">service</tlLemma>
</entry>
<entry id="156">
    <slLemma tag="nocm">ακόλουθος</slLemma>
    <tlLemma tag="nn">attendant</tlLemma>
</entry>
<entry id="157">
    <slLemma tag="nocm">ακουστικό</slLemma>
    <tlLemma tag="nn">hearing</tlLemma>
    <tlLemma tag="nn">aid</tlLemma>
</entry>
<entry id="158">
    <slLemma tag="aj">ακουστικός</slLemma>
    <tlLemma tag="jj">auditory</tlLemma>
```

Fig. 2.3 Sample of the Greek-English dictionary

side, all sentences are split into non-overlapping syntactic phrases using a target language shallow parser. As the proposed methodology has been developed to maximise the use of publicly available software, the user is free to select any desired tools for these pre-processing tasks and there are no restrictions in using any available tool, as long as the developer completes the required integration tasks.

The parallel corpus is stored as two separate XML documents: one containing the tagged-lemmatised SL side and a second one with the tagged-lemmatised and chunked TL side. Samples of these documents can be seen in the extract of a Greek-English parallel corpus in Figs. 2.4 and 2.5. Notably, though the TL side of the corpus is split into phrases, the SL side consists of only the sentence words in sequence, without any information of the corresponding phrases. After the aforementioned preparation of the bilingual corpus, it is passed on to the Phrase Aligner Module for the identification of the corresponding words between SL and TL. The output of PAM in turn is passed to the Phrasing Model Generation module for the production of a parsing scheme on the source side, which will be used to process arbitrary sentences in SL and split them into the corresponding phrases.

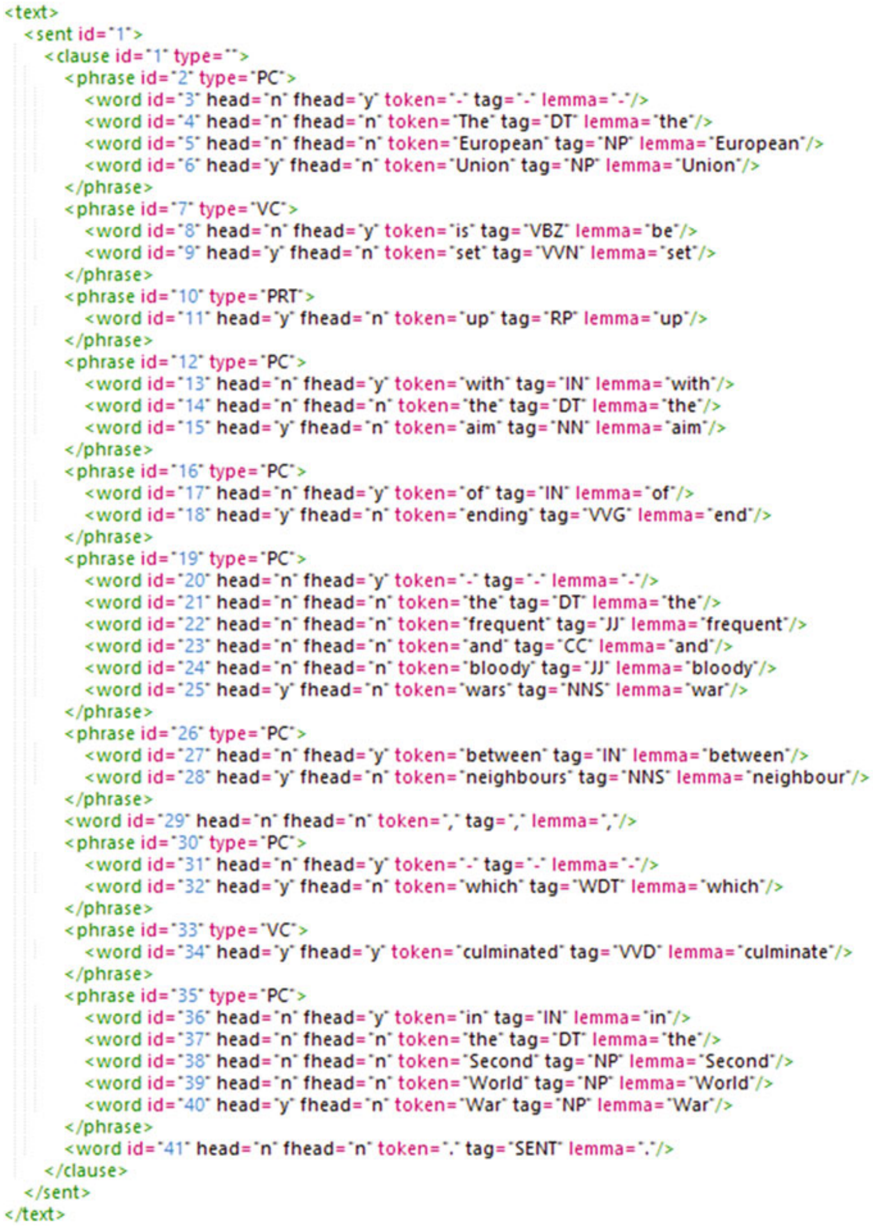


```xml
<text>
  <sent id="1">
    <word id="1" head="n" fhead="n" token="Η" tag="AtDfFeSgNm" lemma="ο"/>
    <word id="2" head="n" fhead="n" token="Ευρωπαϊκή" tag="AjBaFeSgNm" lemma="ευρωπαϊκός"/>
    <word id="3" head="n" fhead="n" token="Ένωση" tag="NoCmFeSgNm" lemma="ένωση"/>
    <word id="4" head="n" fhead="n" token="δημιουργήθηκε" tag="VbMnIdPa03SgXxPePvXx" lemma="δημιουργώ"/>
    <word id="5" head="n" fhead="n" token="με" tag="AsPpSp" lemma="με"/>
    <word id="6" head="n" fhead="n" token="σκοπό" tag="NoCmMaSgAc" lemma="σκοπός"/>
    <word id="7" head="n" fhead="n" token="να" tag="PtSj" lemma="να"/>
    <word id="8" head="n" fhead="n" token="τερματιστούν" tag="VbMnIdXx03PlXxPePvXx" lemma="τερματίζω"/>
    <word id="9" head="n" fhead="n" token="οι" tag="AtDfMaPlNm" lemma="ο"/>
    <word id="10" head="n" fhead="n" token="συχνοί" tag="AjBaMaPlNm" lemma="συχνός"/>
    <word id="11" head="n" fhead="n" token="και" tag="CjCo" lemma="και"/>
    <word id="12" head="n" fhead="n" token="αιματηροί" tag="AjBaMaPlNm" lemma="αιματηρός"/>
    <word id="13" head="n" fhead="n" token="πόλεμοι" tag="NoCmMaPlNm" lemma="πόλεμος"/>
    <word id="14" head="n" fhead="n" token="μεταξύ" tag="AdXxBa" lemma="μεταξύ"/>
    <word id="15" head="n" fhead="n" token="γειτονικών" tag="AjBaFePlGe" lemma="γειτονικός"/>
    <word id="16" head="n" fhead="n" token="χωρών" tag="NoCmFePlGe" lemma="χώρα"/>
    <word id="17" head="n" fhead="n" token="που" tag="PnReMa03PlNmXx" lemma="που"/>
    <word id="18" head="n" fhead="n" token="κορυφώθηκαν" tag="VbMnIdPa03PlXxPePvXx" lemma="κορυφώνω"/>
    <word id="19" head="n" fhead="n" token="με" tag="AsPpSp" lemma="με"/>
    <word id="20" head="n" fhead="n" token="το" tag="AtDfMaSgAc" lemma="ο"/>
    <word id="21" head="n" fhead="n" token="Δεύτερο" tag="NmOdMaSgAcAj" lemma="δεύτερος"/>
    <word id="22" head="n" fhead="n" token="Παγκόσμιο" tag="AjBaMaSgAc" lemma="παγκόσμιος"/>
    <word id="23" head="n" fhead="n" token="Πόλεμο" tag="NoCmMaSgAc" lemma="πόλεμος"/>
    <word id="24" head="n" fhead="n" token="." tag="PTERM_P" lemma="."/>
  </sent>
</text>
```

Fig. 2.4 SL part sample of a Greek-English parallel corpus in PRESEMT

2.2.4 *The TL Monolingual Corpus*

The TL monolingual corpus is significantly larger than the parallel one and can be considered as the main resource for the main translation pipeline, as it is used to build the TL language model responsible for most of the translation tasks. The corpus size is of the order of a billion tokens. For example, the size of the English monolingual corpus used by the Greek-English PRESEMT system contains 3.65 billion tokens, while the size of the German one used by the Greek-German PRESEMT system contains 3.0 billion tokens. However, it should be stressed that the larger size of the monolingual corpus does not represent a constraint to the system creation as for most languages monolingual corpora of a good quality are available and most pre-processing is done offline, to speed up the translation process. All monolingual corpora created in PRESEMT were collected using a Web crawler (Pomikálek and Rychlý 2008). Before they can be used in the system, they are tagged-lemmatised and chunked offline during the pre-processing stage in order to produce the language model. The corpora with all the relevant annotation information are also stored using the PRESEMT XML representation shown in Fig. 2.5.



```
<text>
  <sent id="1">
    <clause id="1" type="">
      <phrase id="2" type="PC">
        <word id="3" head="n" fhead="y" token="-" tag="-" lemma="-"/>
        <word id="4" head="n" fhead="n" token="The" tag="DT" lemma="the"/>
        <word id="5" head="n" fhead="n" token="European" tag="NP" lemma="European"/>
        <word id="6" head="y" fhead="n" token="Union" tag="NP" lemma="Union"/>
      </phrase>
      <phrase id="7" type="VC">
        <word id="8" head="n" fhead="y" token="is" tag="VBZ" lemma="be"/>
        <word id="9" head="y" fhead="n" token="set" tag="VVN" lemma="set"/>
      </phrase>
      <phrase id="10" type="PRT">
        <word id="11" head="y" fhead="n" token="up" tag="RP" lemma="up"/>
      </phrase>
      <phrase id="12" type="PC">
        <word id="13" head="n" fhead="y" token="with" tag="IN" lemma="with"/>
        <word id="14" head="n" fhead="n" token="the" tag="DT" lemma="the"/>
        <word id="15" head="y" fhead="n" token="aim" tag="NN" lemma="aim"/>
      </phrase>
      <phrase id="16" type="PC">
        <word id="17" head="n" fhead="y" token="of" tag="IN" lemma="of"/>
        <word id="18" head="y" fhead="n" token="ending" tag="VVG" lemma="end"/>
      </phrase>
      <phrase id="19" type="PC">
        <word id="20" head="n" fhead="y" token="-" tag="-" lemma="-"/>
        <word id="21" head="n" fhead="n" token="the" tag="DT" lemma="the"/>
        <word id="22" head="n" fhead="n" token="frequent" tag="JJ" lemma="frequent"/>
        <word id="23" head="n" fhead="n" token="and" tag="CC" lemma="and"/>
        <word id="24" head="n" fhead="n" token="bloody" tag="JJ" lemma="bloody"/>
        <word id="25" head="y" fhead="n" token="wars" tag="NNS" lemma="war"/>
      </phrase>
      <phrase id="26" type="PC">
        <word id="27" head="n" fhead="y" token="between" tag="IN" lemma="between"/>
        <word id="28" head="y" fhead="n" token="neighbours" tag="NNS" lemma="neighbour"/>
      </phrase>
      <word id="29" head="n" fhead="n" token="," tag="," lemma=","/>
      <phrase id="30" type="PC">
        <word id="31" head="n" fhead="y" token="-" tag="-" lemma="-"/>
        <word id="32" head="y" fhead="n" token="which" tag="WDT" lemma="which"/>
      </phrase>
      <phrase id="33" type="VC">
        <word id="34" head="y" fhead="y" token="culminated" tag="VVD" lemma="culminate"/>
      </phrase>
      <phrase id="35" type="PC">
        <word id="36" head="n" fhead="y" token="in" tag="IN" lemma="in"/>
        <word id="37" head="n" fhead="n" token="the" tag="DT" lemma="the"/>
        <word id="38" head="n" fhead="n" token="Second" tag="NP" lemma="Second"/>
        <word id="39" head="n" fhead="n" token="World" tag="NP" lemma="World"/>
        <word id="40" head="y" fhead="n" token="War" tag="NP" lemma="War"/>
      </phrase>
      <word id="41" head="n" fhead="n" token="." tag="SENT" lemma="."/>
    </clause>
  </sent>
</text>
```

Fig. 2.5 TL part sample of a Greek-English parallel corpus in PRESEMT

2.3 Processing the Parallel Corpus

The role of the parallel corpus is to produce a phrase-mapping scheme between the SL and TL using suitably chosen lemma and PoS information in both language sides and shallow parsing only in the TL. By only using a chunker in the TL side, PRESEMT avoids the use of an additional external tool, thus increasing portability to new language pairs, while also avoiding potential incompatibilities when creating alignments between words and phrases of the two languages.

The processing is performed in two stages. In the first stage, the TL side parsing scheme is transferred in the SL side by building word and phrase alignments using PAM. PAM transfers the TL side parsing scheme, which encompasses lemma, tag and chunking information (namely phrase boundaries and phrase labels), to the SL side, based on lexical information (retrieved from the lexicon) coupled with statistical data on PoS tag correspondences extracted from the lexicon. PAM follows a three-step process, defining alignments based on (a) lexicon entries, (b) similarity of grammatical features and PoS tag correspondence and (c) the alignments of neighbours of the unaligned words.

In the second stage, an SL phrasing model is constructed by PMG, by applying probabilistic methodologies to the PAM output. This phrasing model is then applied for segmenting any arbitrary SL text being input for translation. Initially, PMG was implemented using Conditional Random Fields (CRF), due to the high representational capabilities of this probabilistic model (Lafferty et al. 2001). Alternative approaches for building PMG based on template-matching principles have been investigated (cf. Tambouratzis et al. 2013), though unless otherwise stated the results reported in this volume utilise the CRF model, which is the default tool used within the PRESEMT methodology.

The following two sections provide a detailed description of (i) the Phrase Aligner and (ii) the Phrasing Model Generation modules, respectively.

2.3.1 Phrase Aligner Module

To determine a model expressing the transfer of phrases from SL to TL, it is essential to have the sentences of the parallel corpus analysed into pairs of corresponding phrases in SL and TL. Development work in the earlier MT system METIS-II (Markantonatou et al. 2009) has demonstrated that when trying to harmonise the phrasings from two independently created parsers/chunkers (where one operates in SL and one in TL), extensive effort is required for their modification to compatible phrasing schemes for SL and TL. Thus, such an approach is not suitable for a methodology intended to be ported to new language pairs with minimal effort. To that end, in PRESEMT the Phrase Aligner Module (Tambouratzis et al. 2011) is developed, to eliminate the need for an SL side parser. PAM is dedicated to transferring to the SL side the TL side parsing scheme, which encompasses phrase boundaries and phrase types. During this transfer, the TL side phrase type is inherited by the corresponding SL phrase. In terms of its two-phase approach, PAM has conceptual similarities to a number of works (cf. Och and Ney 2004; Ganchev et al. 2009), as initially (i) words in the SL sentence are aligned to those of the TL sentence and afterwards (ii) unaligned SL words are grouped into phrases depending on agreement of grammatical features.

PAM follows a process divided into three steps, where in each step the aim is to resolve the alignments for tokens that remain unaligned from earlier steps. In the first step, alignments are performed on the basis of the lemmas included in the bilingual lexicon. Thus, tokens between SL and TL are aligned if the lexicon indicates an equivalence in meanings (i.e. one is a valid translation of the other), provided that there are not multiple ambiguous matches between lexicon entries and SL or TL tokens. Later steps use more general information (such as the PoS tag of the tokens rather than lexical information) to align tokens and thus have a lower likelihood of producing the correct alignment than the first step. In the second step, alignments are determined based on the similarity of grammatical features between adjoining tokens (in morphologically rich languages the information of gender or case agreement may associate to one another related tokens for grouping in the same phrase). Finally, in the third step, unaligned words are aligned based on string similarity as well as on the alignments of their neighbours (under the assumption of locality of alignments). The entire alignment process is described in detail in Tambouratzis et al. (2012).

After the alignments on a token level are completed, the aim becomes to map for each phrase in TL all corresponding tokens in SL so as to create phrases in the SL side of the parallel corpus. This process results in the establishment of the SL side phrases, for each of which a correspondence to a TL phrase is defined.

To establish the SL side phrasing, PAM operates on the parallel corpus by utilising the following resources:

(1) a bilingual lexicon from SL to TL;
(2) an SL tagger-lemmatiser (which may provide both basic PoS characterisation and more refined features, i.e. case, number, person, etc.);
(3) a TL tagger-lemmatiser and shallow parser (which can again provide basic and refined features;
(4) a TL clause boundary detection tool.

Based on this set of inputs, PAM decides on the optimal segmentation of the source sentence into phrases. A multicriterion-type comparison is implemented, where the aforementioned inputs are prioritised and combined. Though not all aforementioned inputs need to be present for PAM to work, their use results in a more accurate alignment.

Alignment Step 1: Lexical information

The bilingual lexicon provides information on likely word and lemma correspondences between SL and TL. The word aligner algorithm performs alignment of SL words to TL words via the bilingual lexicon. The algorithm allows the one-to-one alignment between SL words and TL ones, while rejecting any multiple alignments, unless the lexicon explicitly provides such information. In the case of multiple possible alignments, the principle is that for every word *k* in SL that is potentially aligned to more than one word in TL, the TL word chosen is the one (a) that has the minimum distance from the single-aligned TL word and (b) for which the corresponding single-aligned SL word has the minimum distance in tokens from word *k*.

Correspondences of SL-TL PoS tags are also extracted, by running through the bilingual lexicon to estimate the likelihood of alignment between PoS tags, for instance to extrapolate if a verb in SL is more likely to translate to a verb or to a noun in TL. Such correspondences are used to determine alignments for out-of-vocabulary cases, where the lexicon does not provide sufficient information.

As an example of the PAM operation, let us consider the following pair of sentences:

(*Greek—SL*): *Με την Ευρωπαϊκή Κοινότητα Άνθρακα και χάλυβα αρχίζουν να ενώνονται οι Ευρωπαϊκές χώρες οικονομικά και πολιτικά.*
(*English—TL*): *The European Coal and Steel Community begins to unite European Countries economically and politically.*

It is assumed that the lexicon entries relevant to this pair of sentences are those listed in Table 2.2. Then, in Fig. 2.6, the SL and TL sides are depicted in the form

Table 2.2 Indicative lexicon entries in lemma and PoS tag form

Greek language		English language	
Lexicon entry	PoS tag	Lexicon entry	PoS tag
ευρωπαϊκός	Aj	European	JJ
άνθρακας	No	Coal	NN
χώρα	No	Country	NN

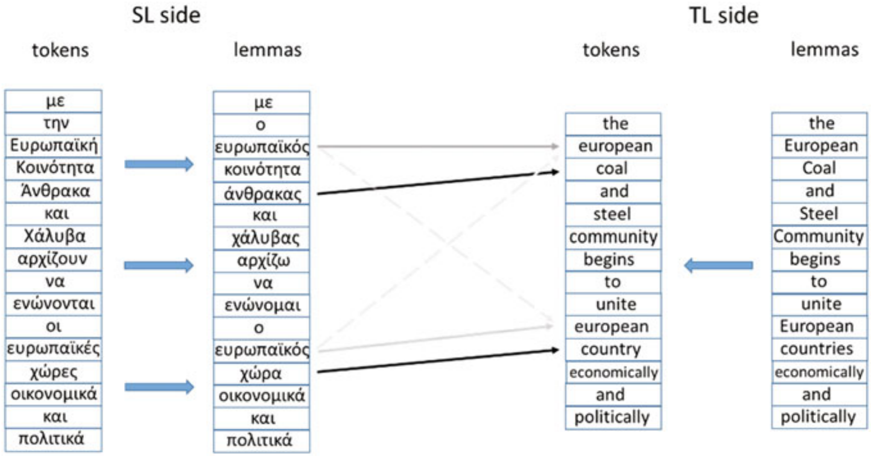


Fig. 2.6 Alignment of words in SL and TL based on the lexical information of Table 2.2

of token and lemma sequences in the left- and right-hand sides, respectively. The alignments that can be identified by the lexicon entries are depicted by arrows from SL to TL, where the dark arrows correspond to unambiguous alignments, while grey arrows indicate more than one possible alignment per SL and TL token.

Based on the entries of Table 2.2, there are unique correspondences in SL and TL for the pairs "άνθρακας"—"coal" and "χώρα"—"country". On the contrary, for the lexicon pair "Ευρωπαϊκός—European", two occurrences exist in both SL and TL and thus two pairs of possible alignments are in consideration (as noted before both SL tokens aligning to the same TL one—or vice versa—are not allowed). In this case, in the absence of any other knowledge, neighbouring tokens for which alignments are already established help to determine the most likely alignments. Thus, the alignments for "coal" and "country" are the preferred ones for the two instances of "European", as indicated by the solid grey arrows in Fig. 2.6, while the less likely alignments are indicated by dashed grey lines in the same figure.

When an SL word remains unaligned, usually due to limited dictionary coverage, the algorithm transliterates it (in case of different SL and TL alphabets, e.g. Greek and English) and consequently attempts to match it to a word with high similarity in the TL sentence. Two words, for which no association is indicated by the lexicon, are considered similar when their letter-wise similarity, in terms of the longest common sub-sequence ratio, exceeds a threshold.

At the end of Step 1, all possible alignments using lexical information have been established. SL words that remain unaligned are handled by subsequent steps using other types of information.

Alignment Steps 2 and 3: Similarity of features

Operating on the output of Step 1, subsequent steps attempt to assign unaligned SL tokens into phrases, by identifying nearby SL tokens which are aligned, and that are similar in terms of grammatical features (as indicated by their extended PoS tags). Thus, for every unassigned SL word the algorithm calculates the similarity of its extended PoS tag with the extended PoS tags of all the already aligned SL words in the sentence. The extended PoS similarity for each word is then normalised by multiplication with a Gaussian function that takes as input the token-wise distance of words on the sentence. Then, PAM clusters words that match to an acceptable extent in terms of tag if they are closely positioned in the sentence.

To illustrate this operation, assume the example shown in Fig. 2.7, regarding a pair of sentences in SL and TL. Several alignments have already been determined via the lexicon correspondences of Table 2.3, as indicated by the black arrows. One of the still unaligned words is "οικονομικά", which is an adjective whose extended tag includes accusative case, neutral gender and plural number. All three characteristics are shared with those of the token "προβλήματα", and thus it is established that these two tokens most likely belong to the same phrase. This alignment is indicated by the dashed arrows of Fig. 2.7.

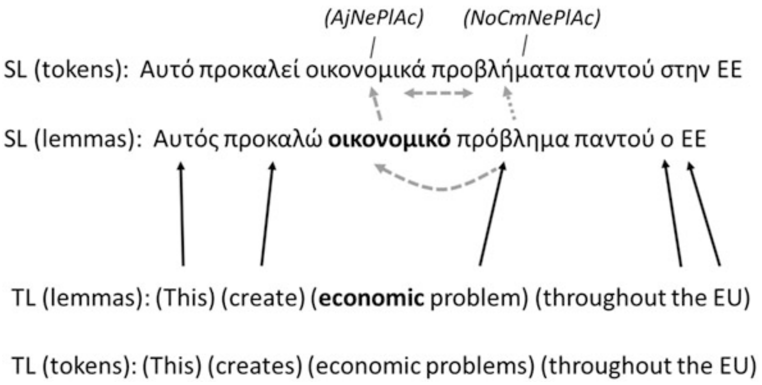

(AjNePlAc)
(NoCmNePlAc)
SL (tokens): Αυτό προκαλεί οικονομικά προβλήματα παντού στην ΕΕ
SL (lemmas): Αυτός προκαλώ οικονομικό πρόβλημα παντού ο ΕΕ
TL (lemmas): (This) (create) (economic problem) (throughout the EU)
TL (tokens): (This) (creates) (economic problems) (throughout the EU)

Fig. 2.7 Example of alignment making use of extended features of neighbouring words (tags are indicated in *brackets* for selected words)

Table 2.3 Lexicon entries relevant to the example of Fig. 2.7

Greek language		English language	
Lexicon entry	PoS tag	Lexicon entry	PoS tag
αυτός	Pn	This	DT
πρόβλημα	No	Problem	NN
ο	At	The	DT
εε	ABBR	Eu	NN

The final alignment of words to this sentence pair is indicated in Fig. 2.8, where the last pending alignments have been resolved. This is achieved by the high likelihood of correspondences within the lexicon entries between tokens with PoS tag “Av” in Greek and tokens with tag “ADV” in English (both of these corresponding to adverbs).

Regarding the bilingual corpora used, it is advisable to edit them so that the SL and TL sides of the corpus are as “close” as possible to each other, removing

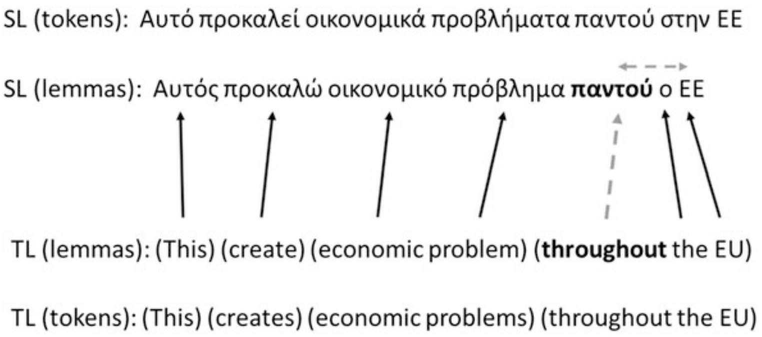

SL (tokens): Αυτό προκαλεί οικονομικά προβλήματα παντού στην ΕΕ
SL (lemmas): Αυτός προκαλώ οικονομικό πρόβλημα παντού ο ΕΕ
TL (lemmas): (This) (create) (economic problem) (throughout the EU)
TL (tokens): (This) (creates) (economic problems) (throughout the EU)

Fig. 2.8 Alignment based on the likelihood of PoS correspondences between SL and TL

metaphors or elliptical constructions and smoothing out divergences between the two languages. In this way, PAM can focus on extracting information about the structural transformations needed to transfer from SL to TL, rather than being affected by divergences between the texts.

2.3.2 *Phrasing Model Generation*

PAM accomplishes the grouping of SL tokens into phrases, in accordance with the given TL parsing scheme. Following the transfer of this phrasing scheme to SL, archetypes become available for developing a phrasing model. This is the task of the Phrasing Model Generation, which learns by example to segment arbitrary input text into phrases in compliance with the TL phrasing scheme. If this is achieved, the aligned parallel corpus can be used to transform the structure of input sentences from SL to TL.

When initiating the work on the Phrasing Model Generation, a survey of relevant work was undertaken to identify appropriate methods. Since this is a widely studied topic, it was decided to select the most promising existing technique (preferably one which employs free-to-use or open-source software), rather than developing or reimplementing a new technique. This can speed up the system development and makes use of already proven techniques (alternative model generation techniques are investigated in Chap. 6). Most relevant studies have converged at using a probabilistic methodology. It is widely accepted that among the statistical-based models used, Conditional Random Fields (Lafferty et al. 2001) provide the most promising avenue for creating parsers (e.g. Sha and Pereira 2003; Tsuruoka et al. 2009). Due to the expressiveness of the underlying mathematical model, CRF requires a large number of training patterns to extract an accurate model. In comparison with other probabilistic modes, CRF has been found to possess a superior performance to both Hidden Markov Models (HMMs) and Maximum Entropy (ME) models by avoiding biasing solutions towards states with few successor states (Wallach 2004).

A small scale experiment was performed prior to proceeding with the implementation of PMG. This experiment compared a custom rule-based system (comprising approx. 10 rules specified by an expert for identifying phrases) to probabilistic models, which was refined over three iterations. It was found that the rule-based system had a segmentation accuracy of just under 70%, much lower than probabilistic models. Among probabilistic models, HMM had an accuracy of 78%, while CRF had an accuracy close to 90%. As a result, CRF was chosen to implement the Phrasing Model Generation module.

The PRESEMT system utilises the CRF model for phrasal segmentation in the SL. One main requirement for the PMG module is to be language-independent, allowing the generation of a model for any language, working on a limited training set. As PRESEMT assumes a parallel corpus of at most a few hundred sentences, the model should be established taking into account the size of the training set.

Thus, one needs to move to a higher level of abstraction, beyond token (or lemma) forms.

Regarding the PMG algorithmic part, the MALLET package (McCallum 2002) was chosen as it is implemented in java, which is also used for the PRESEMT prototype. Different system set-ups were experimentally tested for the CRF model via the options available within MALLET.

Both the default CRF training method (hereafter denoted as “std.”) and the alternative method (denoted as “alt.”) were tested. Both the complete and reduced tagsets (denoted as “std.” and “red.”, respectively) were investigated, to determine the optimal configuration. Another consideration involves the detail used in the sequence of tokens. For each token in the SL side, there exist different levels of representation, at token, lemma and tag information. However, a training set of typically 200 sentences is definitely too small to extract a phrasing model based on lemmas, so it was decided to employ PoS tags to identify phrase boundaries.

Preliminary experimentation showed improved segmentation accuracy when using only the basic Part-of-Speech tag (such as “Vb” for verb or “No” for noun), coupled with the case for tokens that have this type of information. The model order was also varied, this taking into consideration only the information of the present word (model 0) but also the previous one (denoted as model 0-1) or even the two previous ones (model 0-1-2).

Indicative experimental results are reported in Table 2.4, where for each token, the PMG-generated phrasing information is compared to the gold-standard created by a language specialist over a 100-sentence development set. The segmentation accuracy is expressed as the fraction of tokens that are correctly assigned to their corresponding phrases. The CRF phrasing accuracy peaks at 90%, for the reduced tagset including only the PoS tag and the case feature. The best results were achieved when adopting a CRF model with 0-1 order, while higher model orders resulted in no measurable improvement. Hence, a second-order CRF is used as the default parser for the SL side in PRESEMT, using the last two tokens. It should be noted that the MALLET functionalities are integrated within PRESEMT as a separate module. Thus, the user may invoke via two commands the process that creates a new phrasing model, first performing alignment (via PAM) and then generating the phrasing model via the training of CRF.

Table 2.4 PMG experimental accuracies (denoted in percentages)

Feature	Parameters		Model order		
	Tags	Method	0	0-1	0-1-2
1-gram	Std.	Std.	75.4	80.4	77.8
1-gram	Red.	Std.	82.4	88.1	84.4
1-gram	Red.	Alt.	81.3	89.0	86.0
2-gram	Std.	Std.	73.5	74.8	73.3
2-gram	Red.	Std.	85.5	86.7	84.5
2-gram	Red.	Alt.	89.3	90.0	88.7

2.4 Creating a Language Model for the Target Language

The annotated TL corpus is used for the creation of a language model based on syntactic phrases. This model is then applied to the translation pipeline for establishing correct ordering of words within each phrase, disambiguating between alternative translations and handling functional words (for instance, insertion or deletion of articles or negation particles).

Unlike the statistical language models that are based on n-grams of words, the words here are grouped together based on the syntactic phrases extracted from the chunked TL monolingual corpus. All TL phrases are organised in a hash map, using as a key the following three factors: (i) phrase type, (ii) lemma of the phrase head and (iii) PoS tag of the phrase head. Each TL phrase extracted from the corpus is stored in the equivalent hash map along with its number of occurrences in the corpus. Finally, each map is serialised and stored in a separate file in the file system, with an appropriate name for easy retrieval, so that the system will not have to load the whole model in memory during run-time. For instance, for the English monolingual corpus, all verb phrases with the lemma of the head token being "read" and the PoS tag "VV" are stored in the file "Corpora/EN/Phrases/VC/read_VV".

As an example, let us assume a very small TL corpus consisting only of the following sentence: "*A typical scheme would have eight electrodes penetrating human brain tissue; wireless electrodes would be much more practical and could be conformal to several different areas of the brain*". The syntactic phrases extracted from this small corpus are shown in Table 2.5, while the files created for the model are shown in Fig. 2.9. Because all phrases only appear once in the corpus, the frequencies are omitted in the specific example.

It should be noted that, with respect to large corpora, in order to reduce the number of files created, if a TL phrase file remains very small (based on the definition of a small threshold value), i.e. it contains very few frequent phrases (less

Table 2.5 Syntactic phrases extracted from the TL monolingual corpus

ID	Phrase type	Phrase content	Phrase head lemma/PoS
1	PC	A typical scheme	Scheme/NN
2	VC	Would have	Have/VH
3	PC	Eight electrodes	Electrode/NN
4	VC	Penetrating	Penetrate/VV
5	PC	Human brain tissue	Tissue/NN
6	PC	Wireless electrodes	Electrode/NN
7	VC	Would be	Is/VB
8	PC	Much more practical	Practical/JJ
9	VC	Could be	Is/VB
10	PC	Conformal	Conformal/JJ
11	PC	To several different areas	Area/NN
12	PC	Of the brain	Brain/NN

File 1: VC/Have_VH	
2	Would have

File 2: VC/Is_VB	
7	Would be
9	Could be

File 3: VC/penetrate_VV	
2	penetrating

File 4: PC/scheme_NN	
1	A typical scheme

File 5: PC/electrode_NN	
3	Eight electrodes
6	Wireless electrodes

File 6: PC/Tissue_NN	
5	Human brain tissue

File 7: PC/Practical_JJ	
8	Much more practical

File 8: PC/conformal_JJ	
10	conformal

File 9: PC/areas_NN	
11	To several different areas

File 10: PC/brain_NN	
12	Of the brain

Fig. 2.9 Example of monolingual corpus phrases splits into files

Table 2.6 Statistics for the English and German monolingual corpora

	English	German
Size in tokens	3,658,726,327	3,076,812,674
Number of raw text files (*each cont. a block of ca. 1 Mbyte*)	87,000	96,000
Sentence number	1.0×10^8	9.5×10^7
Phrase number	8.0×10^8	6.0×10^8
Number of extracted phrase files	380,000	478,000

than 10 unique ones), it is not stored separately, but its content is moved in a file with similarly rare phrases.

Table 2.6 provides statistics for the phrase TL language models created for the English and German languages.

References

Ganchev K, Gillenwater J, Taskar B (2009) Dependency grammar induction via bitext projection constraints. In: Proceedings of the 47th Annual Meeting of the ACL, Singapore, 2–7 August, pp 369–377

Koehn P (2005) Europarl: a parallel corpus for statistical machine translation. MT Summit 2005, Phuket, Thailand

Lafferty J, McCallum A, Pereira F (2001) Conditional random fields: probabilistic models for segmenting and labelling sequence data. In: 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, pp 282–289

Markantonatou S, Sofianopoulos S, Giannoutsou O, Vassiliou M (2009) Hybrid machine translation for low- and middle-density languages. In: Nirenburg S (ed) Language engineering for lesser-studied languages. IOS Press, pp 243–274

McCallum AK (2002) MALLET: a machine learning for language toolkit. http://mallet.cs.umass.edu

Munteanu DS, Marcu D (2005) Improving machine translation performance by exploiting non-parallel corpora. Comput Linguist 31(4):477–504

Och FJ, Ney H (2004) The alignment template approach to statistical machine translation. Comput Linguist 30(4):417–449

Pomikálek J, Rychlý P (2008) Detecting co-derivative documents in large text collections. In: Proceedings of LREC2008, Marrakech, Morrocco, pp 1884–1887

Prokopidis P, Georgantopoulos B, Papageorgiou H (2011) A suite of NLP tools for Greek. In: Proceedings of the 10th ICGL Conference, Komotini, Greece pp 373–383

Schmid H (1994) Probabilistic part-of-speech tagging using decision trees. In: Proceedings of International Conference on New Methods in Language Processing, Manchester, UK, pp 44–49

Sha F, Pereira FCN (2003) Shallow parsing with conditional random fields. In: Proceedings of HLT-NAACL Conference, pp 213–220

Tambouratzis G, Simistira F, Sofianopoulos S, Tsimboukakis N, Vassiliou M (2011) A resource-light phrase scheme for language-portable MT. In: Proceedings of the 15th International Conference of the European Association for Machine Translation, 30–31 May, Leuven, Belgium, pp 185–192

Tambouratzis G, Troullinos M, Sofianopoulos S, Vassiliou M (2012) Accurate phrase alignment in a bilingual corpus for EBMT systems. In: Proceedings of the 5th BUCC Workshop, held within the LREC-2012 Conference, May 26, Istanbul, Turkey, pp 104–111

Tambouratzis G, Sofianopoulos S, Vassiliou M (2013) Language-independent hybrid MT with PRESEMT. In: Proceedings of HYTRA-2013 Workshop, held within the ACL-2013 Conference, Sofia, Bulgaria, 8 August, pp 123–130

Tsuruoka Y, Tsujii J, Ananiadou S (2009) Fast full parsing by linear-chain conditional random fields. In: Proceedings of the 12th EACL Conference, 30 March–3 April, Athens, Greece, pp 790–798

Wallach HM (2004) Conditional random fields: an introduction. University of Pennsylvania CIS Technical Report, MS-CIS-04-21, February 24

Springer

http://www.springer.com/978-3-319-63105-9

Machine Translation with Minimal Reliance on Parallel Resources
Tambouratzis, G.; Vassiliou, M.; Sofianopoulos, S.
2017, IX, 88 p. 17 illus., Softcover
ISBN: 978-3-319-63105-9