

# A Qualitative Analysis of Moses

Jorik Jonker

July 13, 2008

## Abstract

A state of the art statistical machine translation model is trained, using phrase-based translation and factored models and the Europarl-corpus. This model is used to translate an English text into Dutch, in order to run a qualitative evaluation on it. This evaluation attempts to capture the information preservation during machine translation.

## 1 Introduction

### 1.1 Statistical machine translation

Statistical Machine Translation is the technique of translating by using the mathematics of a text itself. The first steps in this area were set by Weaver (1955), discussing the information-theoretical challenges of machine translation long before “machines” were widely available. A new, statistical approach was introduced by Brown et al. (1993). The whole theory behind statistical machine translation is summarised in equation 1.

$$p(e|f) \tag{1}$$

Equation 1 represents the chance that a string  $e$  in a native language is the translation of the string  $f$ . This probability is estimated using parameter estimation. The problem of translating a string  $f$  into  $e$  is transformed into the problem of finding a string  $e$ , which maximises  $p(e|f)$ :

$$\tilde{e} = \arg \max_{e \in e^*} p(e|f) \tag{2}$$

$$= \arg \max_{e \in e^*} p(f|e)p(e) \tag{3}$$

According to Bayes’ theorem,  $p(e|f)$  can be rewritten as  $p(f|e)p(e)$ , in which  $p(f|e)$  represents the *trans-*

*lation model* and  $p(e)$  the language model. The translation model models the chance that  $e$  is the translation of  $f$  (which is not necessarily the same as  $p(e|f)$ ) and  $p(e)$  models the probability of  $e$ ’s occurrence. As mentioned before, translating  $f$  into  $e$  involves a search for the “best”  $e$ , maximising  $p(f|e)p(e)$ , which is called *decoding*. Decoding always employs heuristics in order to reduce the search space and thus increase efficiency, keeping acceptable quality. Without these heuristics, a decoder would have to use an exhaustive search involving all known native sentences, which is obviously not feasible for most translation tasks.

### 1.2 IBM models

Brown and his team developed a series of five cascading models, often referred as the “IBM models” implementing above theory. Model 1 employs the expectation-maximisation algorithm Dempster et al. (1977) to estimate word to word translations, by counting the co-occurrence of words. Theory is that words with a high translation probability should have a relatively high co-occurrence. A problem with this approach is the re-ordering of words; the translation model developed with model 1 does not account for word reordering. Model 2 address this word reordering, by developing a so called “distortion model”, modelling the displacement of certain words during translation.

A major issue with the first two models is that they are only able to address “literal” translations. Each word is translated into exactly one other word, which becomes a problem when the French phrase *ne pas* would be translated, which means *not*. A third model (model 3) was developed to address this issue, modelling the *fertility* of a word.

A fourth model is introduced, which strongly resembles model 2, since it models the distortion of the words. This time, the introduced fertility is taken into account. Models 3 and 4 are known to have statistical deficiencies.

cies, which results in a lack of convergence guarantee. Model 5 fixes this last issue and provides a definitive “word alignment”. The most common implementation of these models is GIZA++ (Och, 2000), a GPL'd toolkit which is an extension to GIZA, from the EGYPT toolkit.

### 1.3 Word alignment

The parameters of the probabilities in equations 1, 3 and 3 are estimated by word alignments. An alignment is a mapping between foreign and native “language”, in the broadest sense. A good parallel corpus can be considered a sentence aligned document, since each sentence maps to its translation. When a mapping is made from each foreign word to its translation (native), this is called a word alignment. The same can be done with phrases, which is exemplified in figure 1.

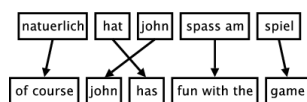


Figure 1: phrase alignment, courtesy of Phillip Koehn

The IBM models mentioned in the previous section ultimately produce a word alignment. By simply counting word co-occurrences, an estimate of word translation probabilities is made. While this may sound very crude, this method proves quite effective given a large enough bilingual corpus (Brown et al., 1993; Marcu and Wong, 2002; Och and Ney, 2003). As Phillip Koehn points out (Koehn and Bojar, 2006), a major drawback of the IBM models is the fact that they produce one to  $n$  mappings: each foreign word can only map to one native word. This problem is addressed by producing an alignment in each translation direction and intersecting these. The intersection of these alignments produces a high-precision alignment, while the union of the two produces additional alignment points. This technique of intersection is shown in figure 2.

### 1.4 Language models

So far, these models have only addressed the problem of providing a translation model. Before decoding, a language model is needed as well. The most often used language model are the  $n$ -gram language models, which are exemplified in figure 3. The idea behind  $n$ -gram language is the fact that the probability of the occurrence

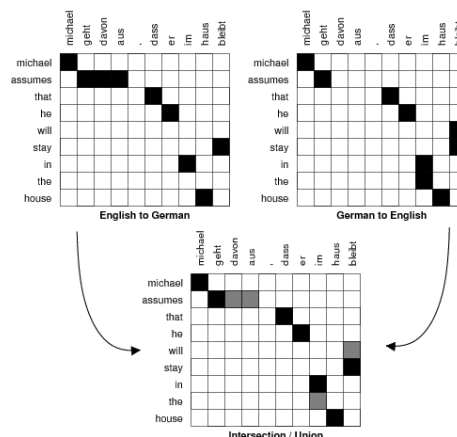


Figure 2: alignment intersection, courtesy of Phillip Koehn

of a sentence  $s = w_1 w_2 \dots w_n$  is equal to the product of the occurrence probability of all  $n$ -grams in that sentence.  $N$ -gram probabilities can conveniently be harvested from corpora by simply counting them.

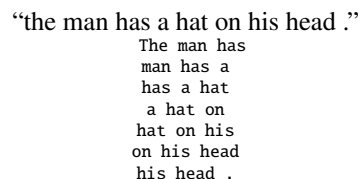


Figure 3: 3-grams, or trigrams

A problem with  $n$ -gram models is the fact that an  $n$ -gram “miss” will normally result in a zero probability. If an  $n$ -gram is never seen before, its occurrence is 0, which will result into a zero sentence occurrence probability ( $x \times 0 = 0$ ). This problem is addressed by a group of techniques called *smoothing*, which estimate the probability of an unseen event. The simplest smoothing technique simply assigns a very low, but non-zero probability to these unseen events. More advanced techniques rely on the availability of another  $n$ -gram model of lower order (lower  $n$ ).

### 1.5 Phase-based

The discussed models so far only align words to words. It is evident that aligning phrases<sup>1</sup> will yield more lin-

<sup>1</sup>Note that the definition of “phrase” is considered as “any sequence of words”

guistic sound results. The work of Marcu (2001) proposes new methods to extract phrase based translation based on the results of the IBM models, by finding contiguous aligned sequences of words. In Marcu and Wong (2002) a different approach is suggested: a joint probability phrase-based model generating the native and foreign sentences in a parallel corpus. Koehn (2003) propose yet another approach, based on the alignment intersection of section 1.3. The phrase alignment is started with the intersection of the word alignments and expanded to a maximum of the union, employing several heuristics.

## 1.6 Factored models

The main “drive” behind traditional statistical machine translation is co-occurrence of translations. While this yields promising results, it has as a big disadvantage that it employs no linguistic knowledge at all. In order to introduce linguistic knowledge into the process of statistical machine translation, Koehn and Bojar (2006); Koehn and Hoang (2007) have designed a novel approach: factored models. In this approach, any additional information (morphological, syntactical, lexical, etc) can be used in order to enhance the translation. This enhancement is done by aligning multiple factors, instead of only the surface words. Each word is a factor by itself, namely the surface form. The part of speech, morphology, lemma, etc. can contribute as a factor too. All factors of a word are joined using a special character, creating a multi-factor corpus, illustrated in figure 4.

$$word_1:pos_1:lemma_1 \ word_2:pos_2:lemma_2 \ \dots \ word_n:pos_n:lemma_n$$

Figure 4: Example of a factored sentence in the corpus

The factored corpus is aligned as a whole, since each underlying factor should align to the same corresponding translating factor. The phrase models are constructed per factor, since the vocabularies differ between factors. The process of translation is split into two parts: the actual *translation* of factors and the *generation* of factors. Figure 5 illustrates that the surface itself is not translated, the lemma is directly translated, the part-of-speech and morphology are translated together, while the surface word is generated from the translated factors.

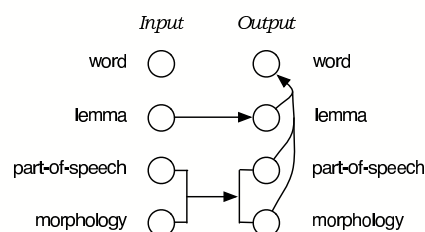


Figure 5: Example of factored translation, courtesy of Phillip Koehn

## 1.7 Decoding

Until now, we only have discussed several aspects of the training process, which is only half of the translating problem. Recapping equation 3 and 3, decoding is the process of actually searching the best translation. The decoder initially segments an input sentence into words or phrases, depending on whether a word- or phrase-based model is used. From each word or phrase, several translation candidates are considered, optimising the sentence translation probability (1). When using a reasonable corpus, the search space of these translation candidates becomes extraordinary huge, making the decoding a dreadful slow process. To enhance this decoding process, several heuristics and optimisations can be made (Germann, 2003; Germann et al., 2001; Knight, 1999). Tillmann and Ney (2003) use a *beam-search* algorithm, using dynamic programming, which constrains word reordering.

## 1.8 Moses

The techniques described in the previous sections is implemented in a software toolkit called MOSES (see Koehn and Bojar, 2006). MOSES essentially is a decoder, but it ships with a set of scripts covering the training process as well. The training of the translation model is done using GIZA++ and MKCLS, an implementation of the work of Och (1999). The language model used in the decoding can either be SRILM (see Stolcke, 2002) or IRST LM, although the former is most commonly used. Moses uses the beam-search algorithm proposed in Tillmann and Ney (2003).

feature	en	nl
words	22.3M	22.1M
sentences	1.1M	1.1M
bytes	120M	130M

Table 1: Basic statistics of the used corpus

## 1.9 Europarl

The corpus used in this research is the Europarl corpus (see Koehn, 2002), which is a multi-language transcript of the meetings of the European parliament. Europarl is a parallel corpus, available in eleven European languages, amongst which English, German, French and Dutch. It comes with a small set of tools, to create a clean, sentence-aligned, tokenised, lowercase parallel text in any combination of the eleven languages. The language pair of choice was defined as English/Dutch, since the author of this research is most familiar with these two. Table 1 shows some basic statistics of this massive bi-text.

### 1.10 Kroon’s evaluation

In his master’s thesis, Kroon (2007) uses a qualitative method for evaluating Brown’s IBM models with the off-the-shelf ISI Rewrite decoder (see Germann, 2003; Germann et al., 2001). Kroon’s evaluation differs from many traditional evaluations such as BLEU (see Papineni et al., 2001), since it addresses the informational content of a text, in stead of lexical correspondence with the “right” translation. Kroon states a BLEU score of 0.3<sup>2</sup> on “his” translation. He uses an official Dutch secondary education exam test (“CITO”) which is translated using IBM and Rewrite. Subjects were faced with either the original text, or the (automatic) translated text. The exam consists of several intrinsic questions about the text, which were scored using the official CITO-procedure. The averaged scores of the original and translation determined the quality of the translation. A key aspect of the quality of this evaluation is that it is performed on both the original as well as the translation, so that the average scores can be compared.

Kroon has done his evaluation using 18 subjects, using an automated “web exam”. Besides the correct answer, he also requires the subjects to indicate whether they have guessed the answer or not. He concludes his research with a precision of 44% on the translated

<sup>2</sup>BLEU scores range from 0 to 1, where 1 is the highest quality

text and 75% on the original, suggesting that during the translation crucial information is lost.

### 1.11 Research question

This research will try to repeat Kroon’s evaluation, but this time based on a different translation framework. Instead of the Rewrite decoder, Moses will be used, using its phrase-based strategy.

Based on the Europarl corpus, do phrase-based factored models produce qualitative better translations than traditional models?

The next chapter will discuss the methodology of this research.

## 2 Methodology

The rough outline of this research is the set of following steps:

1. prepare the corpus;
2. determine training parameters;
3. train the models;
4. decode the foreign text;
5. evaluate the translated text.

The following sections discuss the intrinsics of each step.

### 2.1 Corpus preparation

As discussed in the previous chapter, the Europarl corpus (see Koehn, 2002) will be used, more specific, version 3. Only the language pair English/Dutch is considered.

The corpus needs to be aligned on sentence level first. This can easily done by the tools shipped with the Europarl corpus. Moreover, these tools ensure each sentence is on its own line and that it is properly tokenised<sup>3</sup>. After this sentence alignment, sentences with more than forty or less than three words are dropped (on both sides of the bi-text, of course). Next, all characters are converted to lowercase.

The evaluation of this research requires a corpus as well. The exact same CITO exam used by Kroon will be used, using the same questions.

<sup>3</sup>Each lexical element should be separated from other using spaces

## 2.2 Training parameters

When training the model, several parameters influence the training process. Since the context of this research dictates a tight time frame, most parameters are left to the defaults shipped by GIZA++, mkc1s and Moses. These “non-defaults” are the order and the smoothing of the language model. The language model is of order five, using Kneser Ney smoothing (see James, 2000), after reported success with these models on a web site<sup>4</sup>. An exploratory experiment has shown that it is currently impractical to employ factored models with this language pair (see section 3.2), so we will stick to plain old surface word translation, thus requiring less parameters to tweak. The rest of Moses’ parameters are left default.

## 2.3 Training

Although the language model will be trained using batches and split corpora, this should not result in different results than when training on the whole corpus, according to SRILM’s documentation. All training will be done on a sufficiently fast machine<sup>5</sup> running (Debian) Linux. The training will be invoked using `train-factored-phrase-model.perl`, which is shipped with Moses, since this is a convenient way to launch Moses, GIZA++, mkc1s, which can be a complex job. The following software versions were used:

tool	version
Moses	SVN revision 20080525
SRILM	1.5.6
GIZA++	2.0-gcc41
mkc1s	2.0-gcc41

## 2.4 Decoding

In order to decode an English sentence, Moses will be invoked with the config file produced by the last step of `train-factored-phrase-model.perl`. It is believed that changing these parameters requires a methodology to determine the best parameters and probably a good quantitative metric for automatic parameter optimisation.

<sup>4</sup>[http://guardiani.us/index.php/Moses\\_Language\\_Model\\_Howto\\_v2](http://guardiani.us/index.php/Moses_Language_Model_Howto_v2)

<sup>5</sup>eight Intel 2.0Ghz CPU cores, 4GB memory

## 2.5 Evaluation

The evaluation framework will consist of a web application containing the digital version of Kroon’s CITO exam. A web application is chosen to increase the ease of evaluation, since subjects can be easily reached through digital media and it is relative effortless to respond. The “test” will not contain questions addressing the subject’s confidence on the correctness of the answers (c.f. Kroon), since the interest is focused on “Kroon’s metric” of this translation. Users have a 50% possibility of getting the original English text, or the translated Dutch version. The questions of the CITO exam will be in English, in any case. After five days, the average score per text version (original or translated) will be compared with Kroon’s scores.

## 3 Results

### 3.1 Responses

In the short time frame of this study 13 subjects have responded to cooperate with this research. These subjects had 5 days to respond (and complete) the test, after which the test was closed.

	original	translation
responses	4	9
correct answers	16	19
relative score	80%	42%

Table 2: Results

Table 2 shows the results in terms of correctly answered questions, grouped by text version.

### 3.2 Factored Models

Originally as a part of this study, but later on considered as an exploratory pre-study, an attempt was made to do this research using a factored model. Besides the Europarl corpus, two factors were added:

- part-of-speech tag;
- lemma.

The part-of-speech tag was obtained using Stanford’s tagger (see Toutanova and Manning, 2000; Toutanova et al., 2003), trained on the CGN corpus for the Dutch half of the corpus, with the so called “small tagset”. The

POS of the English part was obtained using the same tagger, with its default training data, which is shipped with the v1.5.1 version of the tool. The lemma of both languages was obtained using the Snowball<sup>6</sup> stemmer, which is shipped with stemming rules for both languages.

Unfortunately, it appeared unfeasible to use the trained data, since the decoder actually crashes when using the training data. It appeared that this crash has something to do with some very “deep” specifics of the way the language model and the decoder are connected to each other. There has been contact with the author of Moses, Philipp Koehn, who was unable to provide help in time.

## 4 Discussion

The results show more or less the same results as Kroon has obtained. His population was somewhat bigger (38%), although both populations were far too small to prefer one of the models pure on statistical basis. Due to the scale of the experiments and the distance between the scores, there is no direct evidence that one of the two translation methods produces qualitative better text than the other. The results may suggest, but it must be stressed that a larger population in both experiments is needed to confirm this, that phrase based statistical machine translation has no obvious advantage over word based models, besides its theoretical comprehensiveness.

One of the reasons why this translation model has such a relative low score is the fact that the translated text and the corpus on which the translator was trained are not particularly in the same domain, although Kroon has attempted to select a text close to the Europarl-domain. Another reason is that translation models have huge amounts of parameters, which could need some tweaking. This tweaking either require extensive knowledge of the corpus and especially the translating software, or a metric to automatically determine model parameters. BLEU is often used as such a metric, but this requires an “official” translation to compare the output to. Moreover, with the training times of this model on the used corpus (which is approximately 2 days), this will take an enormous length of time.

It is believed that the quality of the translation would significantly improve when factored models are used.

<sup>6</sup>see <http://snowball.tartarus.org/>

The main difference between Kroon’s work and this is the employment of phrase-based models and the usage of a different decoder. In a previous exploratory study an attempt was made to setup an evaluation like in this research, using factored models, which has failed unfortunately.

Expectations are high for the factored models, although the (only) software implementation appeared a bit strict on its input. A follow-up experiment in which a factored model is set up to translate English to Dutch using for instance part of speech, lemma and morphology would probably yield better results. Moreover, an experiment like this and Kroon’s using four versions of the text instead of two, namely original, ISI-translated, Phrase-based translated and factored-translated using a far more bigger population could do a definitive comparison of translation models based on information preservation.

## References

- P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311, 1993. ISSN 0891-2017.
- A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.
- U. Germann. Greedy decoding for statistical machine translation in almost linear time. In *Proceedings of HLT-NAACL-2003*, Edmonton, AB, Canada, 2003.
- U. Germann, M. Jahr, K. Knight, D. Marcu, and K. Yamada. Fast decoding and optimal decoding for machine translation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 228–235, Morristown, NJ, USA, 2001. Association for Computational Linguistics.
- F. James. Modified Kneser-Ney Smoothing of n-gram Models. Technical report, RIACS, 2000.
- K. Knight. Decoding complexity in word-replacement translation models. *Computational Linguistics*, 25(4):607–615, 1999. ISSN 0891-2017.
- P. Koehn. Europarl: A Multilingual Corpus for Evaluation of Machine Translation. December 2002.

- P. Koehn. *Noun Phrase Translation*. PhD thesis, University of Southern California, 2003.
- P. Koehn and O. Bojar. Moses – a factored phrase-based beam-search decoder for machine translation. Website, 2006. URL <http://www.statmt.org/moses/>.
- P. Koehn and H. Hoang. Factored translation models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Prague, Czech Republic, June 2007.
- R. W. Kroon. Improving phrase-based translation. Master’s thesis, University of Twente, 2007.
- D. Marcu. Towards a unified approach to memory- and statistical-based machine translation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 386–393, Morristown, NJ, USA, 2001. Association for Computational Linguistics.
- D. Marcu and W. Wong. A phrase-based, joint probability model for statistical machine translation. In *EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 133–139, Morristown, NJ, USA, 2002. Association for Computational Linguistics.
- F. J. Och. An efficient method for determining bilingual word classes. In *Proceedings of the ninth conference of the European chapter of the association for computational linguistics, EACL99*, pages 71–76, Bergen, Norway, June 1999.
- F. J. Och. Giza : Training of statistical translation models, 2000. URL <http://www.fjoch.com/GIZA.html>.
- F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, March 2003. ISSN 0891-2017. doi: 10.1162/089120103321337421.
- K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA, 2001. Association for Computational Linguistics. doi: 10.3115/1073083.1073135.
- A. Stolcke. Srilm - an extensible language modeling toolkit. In *Proceedings of the international conference on Spoken Language Processing*, Denver, Colorado, September 2002. URL <http://www.speech.sri.com/projects/srilm/>.
- C. Tillmann and H. Ney. Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Computational Linguistics*, 29(1):97–133, 2003. ISSN 0891-2017. doi: 10.1162/089120103321337458.
- K. Toutanova and C. D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 63–70, 2000.
- K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 173–180, Morristown, NJ, USA, 2003. Association for Computational Linguistics. doi: 10.3115/1073445.1073478.
- W. Weaver. Translation. In W. N. Locke and A. D. Boothe, editors, *Machine Translation of Languages*, pages 15–23. MIT Press, Cambridge, MA, 1955. Reprinted from a memorandum written by Weaver in 1949.