# A Ngram-based Statistical Machine Translation Approach for Text Normalization on Chat-speak Style Communications

Carlos A. Henríquez Q.
Center for Language and Speech Technologies and Applications
Universitat Politècnica de Catalunya
Barcelona, Spain
carloshq@gps.tsc.upc.es

Adolfo Hernández H.
Center for Language and Speech Technologies and Applications
Universitat Politècnica de Catalunya
Barcelona, Spain
adolfohh@gps.tsc.upc.es

## ABSTRACT

This paper reports our participation on the text normalization shared task campaign organized by the CAW 2.0 workshop. Through a Statistical Machine Translation (SMT) system we managed to produce sentences syntactically correct given sentences written with misspelled words and chatting slangs. This approach was applied on the evaluation campaign's test set to measure its performance. Here the results over a development and test set are analyzed and commented.

## Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text Analysis, Machine Translation*

## Keywords

Text Normalization, Ngram-based Statistical Machine Translation, Language Model, Dictionary

## 1. INTRODUCTION

Over the past few years social networks, chat rooms and forums have become the most important websites for users to share information about their life and interests. This new way of communication has evolved in such a way that they all share a casual common language. The users write on these sites as if they were writing SMS messages on their mobile phones, without paying attention to correct spelling or moreover, using user-created abbreviations for common phrases, e.g. "by the way" is commonly written as "btw".

For these reasons, existing natural language processing tools cannot process the generated content found on these websites. Simple tools like dictionaries are not entirely satisfying, because the same abbreviation may have several expansions with different meanings ("2" could either mean "too", "to" or "two") and a context analysis evaluation should be made to choose the right definition. A machine translation system may address this challenge because it considers both the translation model, which would offer the different meanings for the same abbreviation or misspelled word, and the context analysis, which would consider the current context to choose the best translation.


*CAW2.0 2009*, April 21, 2009, Madrid, Spain.

Among the different machine translation approaches, the statistical Ngram-based system[11] has proved to be comparable with the state-of-the-art phrase-based systems (like the Moses toolkit[8]), as shown in [9] and [4]. The idea for this shared task was to deal with text normalization as a translation task with the Ngram-based system. For this purpose we generated a correct corpus as the target language from a misspelled training text, which served as the source language, in order to build a parallel corpus for the SMT system. This was done using three dictionaries with misspelled words, commonly seen in chat rooms and SMS messages, for replacing the misspelled words into their correct form.

The replacing procedure also used a language model created with a second data set taken from news articles in order to choose the best definition according to its context.

This paper is organized as follows. Section 2 outlines the system, including tuple definition and extraction, translation model and additional feature models, decoding tool and optimization procedure. Section 3 describes the work done on the training dataset previous the generation of the parallel corpus. Section 4 gives details about the generation of our parallel corpus. Section 5 comments about the performance of the SMT system over a development and testing corpus. Section 6 mentions the submissions made to the normalization task. Finally, section 7 sums up the conclusions for this approach.

## 2. NGRAM-BASED SMT SYSTEM

The beginnings of statistical machine translation (SMT) can be traced back from the early fifties; unfortunately, due to the computational limitations at that time, it was not feasible to achieve practical results successfully. During the nineties, a significant increment in both the computational power and storage capacity of computers, and the availability of large volumes of bilingual data, made possible for SMT to become an actual and practical technology.

The first SMT systems were developed in the early nineties [2][3]. For these, translation-model probabilities at the sentence level were approximated from word-based translation models that were trained by using bilingual corpora [3]. In the case of target language probabilities, these were generally trained from monolingual data by using n-grams.

Present SMT systems have evolved from the original ones in such a way that mainly differ from them in two respects:

first, word-based translation models have been replaced by phrase-based translation models [16][10] which are directly estimated from aligned bilingual corpora by considering relative frequencies, and second, the noisy channel approach has been expanded to a more general maximum entropy approach in which a log-linear combination of multiple feature functions is implemented [13].

The translation system employed in this article implements a translation model that has been derived from the finite-state perspective; more specifically, from the work of [5] and [6]. However, whereas the translation model was implemented by using a finite-state transducer, the system presented here implements it using n-grams. In this way, the proposed translation system can take full advantage of the smoothing and consistency provided by standard back-off n-gram models. In addition to the tuple n-gram translation model, the translation system implements specific feature functions that are log-linearly combined along with the translation model for performing the decoding [12].

## 2.1 The tuple n-gram model

The tuple n-gram translation model constitutes the core model implemented by the n-gram-based SMT system. As already mentioned, the translation model implemented in this work is based on bilingual n-grams, where translation model probabilities at the sentence level are approximated as described by the following equation:

$$P(E,F) \approx \prod_{k=1}^{K} p((e,f)_k | (e,f)_{k-1}, (e,f)_{k-2}, \ldots, (e,f)_{k-n+1}) \quad (1)$$

where $e$ refers to target, $f$ to source, and $(e,f)_k$ to the $k^{th}$ tuple of a given bilingual sentence pair. It is important to note that since both languages are linked up in tuples, the context information provided by this translation model is bilingual.

Tuples are extracted from a word-to-word aligned corpus in such a way that a unique segmentation of the bilingual corpus is achieved. Although in principle any Viterbi alignment should allow for tuple extraction, the resulting tuple vocabulary depends highly on the particular alignment set considered, and this impacts the translation results. For this evaluation, we used a alignment set known as "grow-diagonal-final-and"[10], which is built starting with the intersection set and adding links from the union set until some requisites are accomplished.

In this way, as opposed to other implementations, where one-to-one [1] or one-to-many [6] alignments are used, tuples are extracted from many-to-many alignments. This implementation produces a monotonic segmentation of bilingual sentence pairs, which allows to capture contextual information into the bilingual translation unit structures. This segmentation also allows for estimating the n-gram probabilities appearing in (1). In order to guarantee a unique segmentation of the corpus, tuple extraction is performed according to the following constraints [7]:

1. A monotonic segmentation of each bilingual sentence pair is produced.

2. No word inside the tuple is aligned to words outside the tuple.

3. No smaller tuples can be extracted without violating the previous constraints.

Notice that, according to this, tuples can be formally defined as the set of shortest phrases that provides a monotonic segmentation of the bilingual corpus.

Figure 1 shows an example of sentence pair segmentation using word-to-word alignments from which seven sequential segments (tuples) are extracted.

## 2.2 Log-linear combination framework

The translation system implements a log-linear model where a foreign language sentence $f_1^J = f_1, f_2, ..., f_J$ is translated into another language $e_1^I = e_1, e_2, ..., e_I$ by searching for the translation hypothesis $\hat{e}_1^I$, maximizing a log-linear combination of several feature models [2]:

$\hat{e}_1^I = \arg\max_{e_1^I} \left\{ \sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J) \right\}$

where the feature functions $h_m$ refer to the system models and the set of $\lambda_m$ refers to the weights corresponding to these models.

The translation system for CAW2.0, besides the bilingual translation model, which consists of a 4-gram language model of tuples with Kneser-Ney discounting (estimated with SRI Language Modeling Toolkit[15][1]), implemented a log-linear combination of two additional feature models:

- A **target language model** (a 5-gram model of words, estimated with Kneser-Ney smoothing). This feature provides information about the target language structure and fluency. It favors those partial-translation hypotheses that are more likely to constitute correctly structured target sentences over those that are not. The model is implemented by using a word n-gram model of the target language. This model only depends on the target side of the data, and was trained with additional information from other available monolingual corpora.

- A **brevity penalty model**. This feature introduces a bonus that depends on the partial translation hypothesis length, which is used to compensate the system's preference for short output sentences. The model is implemented through a bonus factor that directly depends on the total number of words contained in the partial-translation hypothesis.

Decisions on the particular translation model configuration and smoothing technique were taken on the minimal-perplexity and maximal-BLEU [14] bases.

## 2.3 N-gram based decoding

The decoder (called MARIE), an open source tool[2], implements a beam search strategy with distortion capabilities was used in the translation system.

The decoding is performed monotonically and is guided by the source language. During decoding, partial-translation hypotheses are arranged into different stacks according to the total number of source words they cover. In this way, a given hypothesis only competes with those hypotheses that provide the same source-word coverage. At every translation step, stacks are pruned to keep decoding tractable.

[1] http://www.speech.sri.com/projects/srilm
[2] http://gps-tsc.upc.es/veu/soft/soft/marie

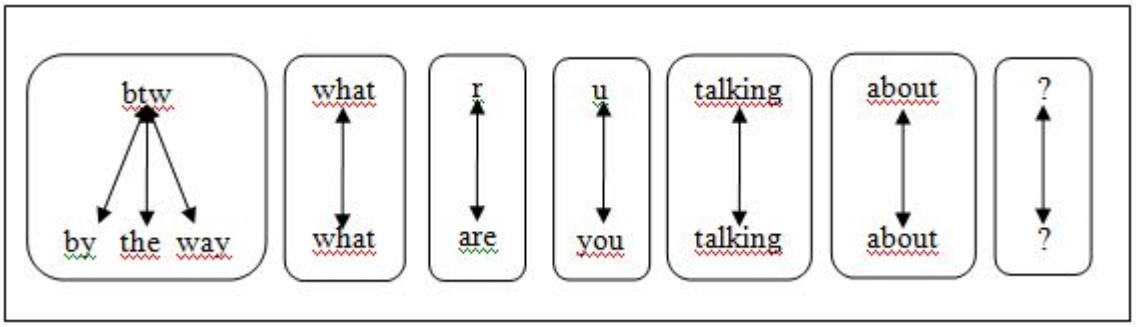


**Figure 1: Segmentation into tuples of an aligned sentence pair**

| | |
|---|---|
| Number of sentences | 74,343 |
| Max. sentence size | 153 |
| Min. sentence size | 1 |
| Avg. sentence size | 21.27 |
| Running words | 1,580,972 |
| Vocabulary size | 86,763 |

**Table 1: Statistics from the News Commentary corpus**

Additionally, MARIE allows for hypothesis recombination, which provides a more efficient search. MARIE also allows for considering additional feature functions during decoding. All these models are taken into account simultaneously, along with the n-gram translation model.

## 2.4 Optimization

Once the models are computed, a set of optimal log-linear coefficients is estimated via an optimization procedure. First, a development data set which does not overlap neither the training set nor the test set is required. Then, translation quality over the development set is maximized by iteratively varying the set of coefficients. In our SMT implementation, this optimization procedure is performed by using a simplex optimization method (with the optimization criteria of the highest BLEU score).

# 3. CAW2.0 2009 EVALUATION FRAMEWORK

For the text normalization shared task, a training dataset taken from six different public sites with different writing styles was provided by the organizers. Because of the different sources and styles, the dataset contains information about the user who wrote it and the thread he was commenting about. Only the text with the message written by the user was considered to build the SMT system.

Besides the dataset provided by the evaluation campaign, the News Commentary corpus (used on 2009 ACL's WMT international evaluation campaign[3]) was used to build a language model of syntactically correct English. The main statistics of this additional corpus are shown in Table 1.

Finally, a dictionary was created crawling through three web pages[4] that provide information about common abbreviations found on chats, forums and SMS messages.

[3]http://www.statmt.org/wmt09/

[4]http://www.alphadictionary.com/articles/imglish/index.html

## 3.1 Building the Dictionary

To create the dictionary, the obtained information was lowercased and joined, removing repeated terms with the same definition. Nevertheless a term may hold several definitions. Finally, a dictionary with 1,630 terms was built.

The dictionary was used to learn possible translations of misspelled words. Named entities and other terms that could be out of vocabulary (OOV) during translation remained unchanged.

## 3.2 Preprocessing the Training Dataset

To build the training corpus, the provided dataset was processed in the following way:

1. Only $<$ *body* $>$ tag content was extracted from the dataset.
2. The text was lowercased, HTML characters and some word forms were amended.
3. Words with asterisks were "normalized" to contain only two (e.g. "f***" and "f*d" were replaced by "f**" and "f**d" respectively).
4. Lines were split after punctuation marks (dot, question and exclamation marks).
5. The final text was tokenized.

A final filtering step was applied to the training corpus. This process kept only the sentences that would be affected by the dictionary. The rest of the sentences did not provide valuable information to the SMT system and therefore were safely removed. The resulting training corpus was the "source language" of our SMT system. This step left us with a dataset of 400,000 lines, which represents 10% of the original corpus. Table 2 describes the main statistics of the corpus before and after the filtering process.

From this corpus we extracted 1500 sentences for tuning the SMT system and 1500 sentences for testing it as commented in section 2.4.

# 4. PARALLEL CORPUS GENERATION

In order to build the SMT system, the"target language" was generated using the dictionary and the News Commentary corpus. This additional corpus was used to build both a

http://www.smsdictionary.co.uk/abbreviations
http://www.qq.co.za/chatterms.aspx

| | Before filtering | After filtering |
|---|---|---|
| Number of sentences | 4,083,302 | 460,990 |
| Max. sentence size | 8,851 | 100 |
| Min. sentence size | 1 | 1 |
| Avg. sentence size | 13.39 | 14.39 |
| Running words | 54,687,940 | 6,633,918 |
| Vocabulary size | 833,623 | 209,237 |

**Table 2: Statistics from the provided corpus before and after the filtering process**

| | Before tuning | | After tuning | |
|---|---|---|---|---|
| | News+Web | News | News+Web | News |
| BLEU | 97.90 | 96.55 | 99.61 | 99.72 |
| WER | 1.7975 | 2.6926 | 0.2885 | 0.2108 |

**Table 3: Translation quality over development corpus**

vocabulary set and a language model of correct written English. The language model of order 5 (a typical size on Machine Translation) was built using the SRI Language Modeling toolkit.

The algorithm to replace the terms that appear in the source language by their correct spelling form was the following:

- If a word $X$ was not in the vocabulary and it had only one definition in the dictionary, it would be replaced automatically by the corresponding definition.
- If a word $X$ had more than one definition, then the language model was applied over a string consisting of $X$'s three previous words, all the words of a given definition of $X$ and the next two words after $X$. The definition with the best probability was the one that replaced $X$.
- All words that appeared in the vocabulary were not replaced, even though they could appear in the dictionary.
- Sentences that contains more than 100 words were split because of a constraint on the alignment process.

# 5. NGRAM-BASED SMT PERFORMANCE

Two different development procedures were performed in order to see the effect of the target language model. The first one used a language model built with the News Commentary Corpus and the training corpus; the latter only used the News Commentary Corpus. Table 3 shows the translation quality before and after the optimization, Table 4 shows the translation quality over the test set.

| | News+Web | News |
|---|---|---|
| BLEU | 99.53 | 99.61 |
| WER | 0.3311 | 0.2796 |

**Table 4: Translation quality over test corpus**

It can be seen in Table 4 that adding the training corpus to build the language model did not improve the translation quality nor the Word Error Rate (WER) as much as without it. Therefore it was better to use only a syntactically correct corpus to build the language model, even though it belongs to a different domain.

# 6. SUBMISSIONS

Because of the better results that gave the SMT system with the language model built only with the News Commentary Corpus, this approach was the one submitted as the primary transcription for the shared task. An alternative transcription obtained with the strategy used to build the parallel corpus (section 4) was also submitted. The idea is to compare the effects of context information learned by the SMT system with the two words context information provided to the corpus generation approach.

# 7. CONCLUSIONS

In this paper we introduced an approach based on a Ngram-based SMT system to participate on the CAW 2.0 evaluation for the text normalization task. Apart from summarizing the SMT system, we have presented the feature models that were taken into account, along with the bilingual Ngram translation model. We also showed the generation of the parallel corpus based on language modeling and dictionaries.

This approach had a strong dependency on the dictionary quality and size. A small dictionary is not able to handle all possible abbreviation and terms.

To show the performance of the SMT approach, the following experiments with the test set delivered by the organizers were done: text normalization using the SMT system and text normalization only using the algorithm discussed in section 4, i.e. using a dictionary and the correct English language model.

# 8. REFERENCES


[1] S. Bangalore and G. Riccardi. Stochastic finite-state models for spoken language machine translation, 2000.

[2] P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, 1990.

[3] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[4] C. Callison-Burch, C. Fordyce, P. Koehn, C. Monz, and J. Schroeder. (meta-) evaluation of machine translation. In *Proceedings of ACL-2007 Workshop on Statistical Machine Translation*, pages 136–158, 2007.

[5] F. Casacuberta. Finite-state transducers for speech input translation. In *Proc. IEEE ASRU*, Madonna di Campiglio, Italy, 2001.

[6] F. Casacuberta and E. Vidal. Machine translation with inferred stochastic finite-state transducers. *Computational Linguistics*, 30(2):205–225, 2004.

[7] J. M. Crego, J. Mariño, and A. de Gispert. Finite-state-based and phrase-based statistical

machine translation. *Proc. of the 8th Int. Conf. on Spoken Language Processing, ICSLP'04*, pages 37–40, October 2004.

[8] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the ACL 2007*, 2007.

[9] P. Koehn and C. Monz. Manual and automatic evaluation of machine translation between european languages. In *Proceedings on the HTL-NAACL Workshop on Statistical Machine Translation*, 2006.

[10] P. Koehn, F. Och, and D. Marcu. Statistical phrase-based translation. In *Proc. of the 41th Annual Meeting of the Association for Computational Linguistics*, 2003.

[11] J. B. Mariño, R. E. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. A. Fonollosa, and M. R. Costa-jussà. N-gram based machine translation. *Computational Linguistics*, 32(4):527–549, 2006.

[12] J. Mario, R. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. Fonollosa, and M. Ruiz. Bilingual n-gram statistical machine translation. In *Proc. of Machine Translation Summit X*, pages 275–82, Phuket, Thailand, 2005.

[13] F. Och and H. Ney. Dicriminative training and maximum entropy models for statistical machine translation. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, Philadelphia, PA, July 2002.

[14] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. IBM Research Report, RC22176, September 2001.

[15] A. Stolcke. SRILM: an extensible language modeling toolkit. In *Proc. of the Int. Conf. on Spoken Language Processing*, pages 901–904, Denver, CO, 2002.

[16] R. Zens, F. Och, and H. Ney. Phrase-based statistical machine translation. In S. Verlag, editor, *Proc. German Conference on Artificial Intelligence (KI)*, september 2002.