Investigations on Machine Translation System Combination

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der RWTH Aachen University zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Diplom–Informatiker Markus Freitag

aus Radolfzell, Deutschland

Berichter:
Prof. Dr.-Ing. Hermann Ney
Prof. Dr. François Yvon

Tag der mündlichen Prüfung: 6. April 2016

Abstract

Machine translation is a task in the field of natural language processing whose objective is to translate documents from one human language into another human language without any human interaction. There has been extensive research in the field of machine translation and many different machine translation approaches have emerged. Current machine translation systems are based on different paradigms, such as e.g. phrases, phrases with gaps, hand-written rules, syntactical rules or neural networks. All approaches have been proven to perform well on several international evaluation campaigns, but no one has emerged as the superior approach. In this thesis, we investigate the combination of different machine translation approaches to benefit from all of them.

The combination of outputs from multiple machine translation systems has been successfully applied in state-of-the-art machine translation evaluations for several years. System combination is a reliable method to combine the benefits of different machine translation systems into one single translation output. System combination relies on the concept of majority voting and the assumption that different machine translation engines produce different errors at different positions, but the majority agrees on a correct translation. Confusion network decoding has emerged as one of the the most successful approaches in combining machine translation outputs. The main goal of this thesis is to develop novel methods to improve the translation quality of confusion network system combination.

In this thesis, we introduce a novel system combination implementation which has been made available as open-source toolkit to the research community. We extend previous invented approaches by the addition of several models and show that our methods produce better or similar translation results as the previous invented approaches. Moreover, compared to one single system combination approach, our implementation is significantly better in several translation tasks.

On top of this high-level baseline, we extend the confusion network approach with an additional model learned by a neural network. The system combination output is typically a combination of the best available system engines and ignores the output of weaker translation systems, although they could be helpful in some situations. We show that our novel model also takes weaker systems into account and detects the positions where the weaker systems help to improve the quality of the combined translation.

One of the most important steps in system combination is the pairwise alignment process between the different input systems. We introduce a novel alignment algorithm which is based on the source sentence and improves the translation quality of our combined translation. In addition to automatic evaluations, we also let humans evaluate our novel approach.

Furthermore, we investigate the effect of decoding direction in the commonly used phrase-based and hierarchical phrase-based machine translation approaches. We show how to benefit from system combination and combine different machine translation setups that are based on different decoding directions. In addition, we investigate techniques to combine the different configurations in an earlier stage, e.g. after the alignment training or the phrase extraction step.

Finally, we present our recent evaluation results that were obtained with our previously invented methods. We participated in the most recent international evaluation campaigns and demonstrate that our methods outperform the translation setups of all participating top-ranked international research labs in several language pairs.

Zusammenfassung

Maschinelle Übersetzung ist ein Teilgebiet der Verarbeitung natürlicher Sprachen, dessen Aufgabe es ist, ohne menschliche Interaktion ein Dokument aus einer gesprochenen Sprache in eine andere gesprochene Sprache zu übersetzen. In dem Gebiet der maschinellen Übersetzung wurde bisher viel geforscht und es entwickelten sich viele verschiedene Übersetzungsansätze. Aktuelle maschinelle Übersetzungssysteme basieren auf verschiedenen Herangehensweisen, wie z.B. Phrasen, Phrasen mit Lücken, handgeschriebenen Regeln, syntaktischen Regeln oder neuronalen Netzen. Bei mehreren internationalen Evaluationskampagnen wurde bewiesen, dass alle diese Methoden gut funktionieren, jedoch kein Ansatz den anderen deutlich überlegen ist. In dieser Doktorarbeit wird die Kombination verschiedener maschineller Übersetzungssysteme untersucht, um von den Vorteilen aller Methoden zu profitieren.

Systemkombination ist eine bewährte Methode, um die verschiedenen Übersetzungsansätze zu kombinieren. Sie beruht auf dem Konzept der Stimmenmehrheit und der Annahme, dass an jeder Position unterschiedliche Übersetzungssysteme Fehler produzieren, die Mehrheit jedoch diese Position fehlerfrei übersetzt. Confusion network decoding hat sich als einer der erfolgreichsten Ansätze zum Kombinieren verschiedener Übersetzungssysteme herausgestellt. Das Hauptziel dieser Doktorarbeit ist die Verbesserung dieser Kombinationsmethode.

Als Teil dieser Arbeit stellen wir eine neue Implementierung vor, welche anderen Forschern als open-source Toolkit frei zur Verfügung gestellt wird. Wir erweitern die bereits erfundenen Kombinationsmethoden und zeigen, dass unser Ansatz gleiche oder bessere Übersetzungsergebnisse liefert. Zusätzlich führen wir ein weiteres Modell ein, dass mit einem neuronalen Netz trainiert wird. Die kombinierte Übersetzung wird hauptsächlich aus den besten Systemen gebaut. Schwächere Systeme werden meist ignoriert, obwohl sie an vielen Stellen hilfreich sein könnten. Unser neues Modell berücksichtigt auch schwächere Systeme und verwendet diese, um die Übersetzungsqualität zu steigern.

Einer der wichtigsten Schritte in der Systemkombination ist die wortweise Alignierung der einzelnen Übersetzungsalternativen. Wir führen einen neuen Alignierungsalgorithmus ein, der zusätzlich zu den Übersetzungen auch deren Quellsätze mit einbezieht. Zusätzlich zu der Verbesserung der automatisch berechneten Fehlermaße zeigen wir, dass auch Menschen unsere neuen Übersetzungen bevorzugen.

Wir untersuchen die Decodierungsrichtungen zweier weit verbreiteter Übersetzungsansätze, die entweder auf Phrasen oder zusätzlich auf Phrasen mit Lücken basieren. Wir untersuchen, wie die verschiedenen Decodierungsrichtungen mit unseren zuvor entwickelten Methoden kombiniert werden können. Als Abschluss dieser Arbeit präsentieren wir die aktuellsten Evaluationsergebnisse, die mit den Algorithmen aus dieser Doktorarbeit erzielt wurden.

Acknowledgements

Although the last five years (of plenty free evenings and weekends) were tough, I finally come to the point to thank everyone who guided and accompanied me along the way.

First, I would like to thank Prof. Dr. Hermann Ney for giving me the opportunity to work at the i6. I learned a lot during my time at the institute not only from him, but also from his former PhD students, which went through his training camp. Having the opportunity to attend so many conferences to meet lots of interesting people was very valuable.

I would also like to thank Prof. François Yvon to be my second supervisor and taking the time to give me valuable feedback.

I definitely want to give a special thank to Stephan Peitz. My former roommate, who always had time for discussing any crazy, weird new ideas. Not only related to statistical machine translation, but also related to any other aspects in life. You know my PhD was only one bullet on my list. Thanks for pushing me all the time. Btw, ist das jetzt ein halber oder ein ganzer Liter Milch?

I would like to deeply thank Joern Wuebker, Matthias Huck, and Minwei Feng who all had an special impact on my research results. I would also like to thank Malte Nuhn, Saab Mansour, Tamer Alkhouli, JTP, Andy Guta, Patrik Lehnen, Christoph Schmidt, and Gregor Leusch for working with me on machine translation. I enjoyed being with you on several conferences around the world. It was a pleasure to work with all of you guys.

Thanks also for the nice coffee breaks and discussions with Jens Forster, Martin Sundermeyer, Markus Nussbaum, Oscar Koller (+ the pleasure to commute together), Stefan Koltermann, Zoltan Tuske, Yannick Gweth, Carmen Heger, Simon Wiesler, David Nolden, Pavel Golik, Harald Hanselmann, Albert Zeyer, Markus Kitza, Kai Frantzen, Christian Plahl, Arne Mauser, David Rybach, and many more.

Big thanks also to my two former Diploma thesis supervisors David Vilar and Daniel Stein, who introduced me to the theory of machine translation and this institute.

I would also like to thank Yaser, Graham, Abe, Martin, and all the other IBM guys for having the opportunity to work with you during my internship in 2013.

Last but definitely not least, I want to thank my family for supporting me during this time. My gorgeous wife Sarah has been a tremendous partner all over the years. Your love, your loyalty, and your support is of inestimable value for me. My two sons Ben and Eric, who always remind me about the importance of life. I will give my very best to guide you through your entire life. I will always be there for you and make sure that all your dreams come true.

A special thank goes to my mother, who always supports me in any aspects of life. Thank you for all the opportunities you gave me in the last 30 years. I would also thank my sister Cathrin for her support. It has been a pleasure to learn Latin with you :)

Second Star to the Right and Straight on Till Morning

Peter Pan

Contents

1 Introduction

Machine translation is the challenge of translating sentences from one language to another language without any human assistance. The first known machine translation ideas have been proposed in the 1950s by [Bar-Hillel 51, IBM 54]. Since then many different machine translation approaches emerged, but even the best automatic produced translations are often ungrammatical and need post-editing for better understanding.

Nevertheless, there are several applications for partly correct translations. For example customer reviews of products sold in the wide world web are very useful for potential buyers. Instead of just providing the reviews written in one language, Internet companies provide additionally automatic translated reviews originally written in a foreign language. The user is more interested in the meaning and less in the grammatical correctness of the translation. Nowadays, Internet companies adapt their translation engines for their specific need of application and yield better translation quality on their specific domain.

Even in areas where high-quality translations are needed, machine translation is helpful. Human translators use the automatic translations as a first draft for their final translations. The human translator revises only the errors occurred through the automatic translation process. In doing so, the high-quality translation process is faster and hence cheaper. In a comparable approach, the human translators are only given partly translated sentences. These segments are extracted from previous translated sentences and hence correct translations. On top of these correct translated segments, the human translator translates the missing parts.

In the last years, there has been extensive research in machine translation. Different machine translation approaches emerged which all of them have their own strengths and weaknesses. Combining different approaches to yield better translation quality is the task of machine translation system combination. In system combination the output of different machine translation engines or/and different setups of the same engine are combined to produce a new combined output which is better compared to all single engines.

1.1 Approaches for Machine Translation

In the last years, different machine translation approaches have been emerged. Two of the most successful approaches are the *rule-based translation* and the *statistical data-driven translation* approach.

In a rule-based translation system, the source language sentence is analyzed with morphological tools, part-of-speech taggers, syntax parsers, etc.. Based on this information, the sentence is transformed into an intermediate representation from which the target sentence is generated by hand-written rules generated by human language experts. For each language pair, a new set of rules is needed to produce a translation output.

The statistical data-written approach is based on a large set of previously translated documents called bilingual data and on an additional large set of documents in the target language called monolingual data. Given these documents, statistical models are calculated which assign all possible translations a probability to be the translation of a given source sentence. In most recent evaluation campaigns, the statistical approach outperforms the rule-based approach. In this thesis, we focus on the statistical approach. More details are given in chapter 3. The statistical machine translation approach can be further divided into different sub approaches which all have in common to rely on statistical calculated probability models.

1.2 System Combination

System combination for machine translation is an area of research which has demonstrated significant gains in translation quality over the traditional translation approaches. The idea is to combine different translation outputs into a stronger one. A number of combination schemes have appeared in the literature, most of them are modified versions of the original ROVER [Fiscus 97] scheme that has been developed for combining speech recognition systems. Probably the most important requirement for successfully combining different machine translation approaches is that of system diversity; as long as systems make independent errors when they translate a source sentence, system combination has a good chance of correcting those errors and improving the translation quality. As part of this thesis, we develop new methods for combining several machine translation systems into a stronger one.

1.3 Outline

In this thesis we present, analyze, and extend the confusion network system combination approach. System combination is a method for combining the automatic generated translation outputs of different machine translation engines into a stronger one. As in the previous section mentioned, there are several different machine translation approaches which all have their advantages and disadvantages. The challenge of system combination is to bring all of the different advantages in one stronger translation.

We start to formulate the scientific goals of this thesis in Chapter 2. We introduce the theory of statistical machine translation in Chapter 3. In Chapter 4, a novel system combination implementation is introduced which is part of the open-source statistical machine translation toolkit `Jane` and has been developed as part of this thesis. The theory presented in this chapter is essential to understand the rest of this thesis. In Chapter 5, we extend the previously introduced system combination approach by an additional model learned by a neural network. A different extension based on alignments to the source sentence is presented in Chapter 6. In Chapter 7, the decoding directions of standard phrase-based and hierarchical phrase-based decoders are investigated. In Chapter 8, the recent evaluation results conducted with the previously invented methods are presented. A conclusion of the thesis is given in Chapter 9. An overview about the main chapters is given in Table 1.1.

Table 1.1: Overview of the main chapters.

Chapter	Topic
Chapter 4	Novel confusion network system combination approach.
Chapter 5	Additional confusion network system combination model learned by a neural network.
Chapter 6	Novel system combination approach based on source-to-target phrase alignments.
Chapter 7	Investigation of decoding directions for statistical machine translation and its benefits for system combination.

1.4 Publications

Most of the work presented in this thesis has been published in several international scientific conferences and journals. Below is given a list of all publications that have been successfully submitted during the work on this thesis. Most of the papers are joint work with several colleagues. A short statement is added to each paper which describes my part in each of the publications.

- System Combination:
  - The following papers are based on my idea, implemented by myself and all experiments have been fully ran by myself:
    - [Freitag & Peter$^+$ 15] Local System Voting Feature for Machine Translation System Combination
    - [Freitag & Huck$^+$ 14] Jane: Open Source Machine Translation System Combination
    - [Freitag & Wuebker$^+$ 14] Combined Spoken Language Translation
    - [Freitag & Feng$^+$ 13] Reverse Word Order Models
  - The following papers are joint work with project partners. I did all system combination experiments, the single engines have been generated by the different partners. All papers include novel system combination extensions which have been developed by myself:
    - [Freitag & Peitz$^+$ 14] EU-BRIDGE MT: Combined Machine Translation
    - [Freitag & Peitz$^+$ 13] EU-BRIDGE MT: Text Translation of Talks in the EU-BRIDGE Project
    - [Freitag & Peitz$^+$ 12] Joint WMT 2012 Submission of the QUAERO Project
    - [Freitag & Leusch$^+$ 11] Joint WMT Submission of the QUAERO Project
  - I provided the implementation for this paper:
    - [Feng & Freitag$^+$ 13] The System Combination RWTH Aachen: SYSTRAN for the NTCIR-10 PatentMT Evaluation
  - I did half of the experiments for our group and compared the GIZA++ alignment with the other ones:
    - [Rosti & He$^+$ 12] Review of Hypothesis Alignment Algorithms for MT System Combination via Confusion Network Decoding
  - I did everything of the German→English section:
    - [Leusch & Freitag$^+$ 11] The RWTH System Combination System for WMT 2011

- Machine Translation Research:
  - Extension of previous work. Adaptation of the previous tools to the phrase-based decoder.
    - [Wuebker & Huck$^+$ 12] Jane 2: Open Source Phrase-based and Hierarchical Statistical Machine Translation
    - [Huck & Peter$^+$ 12] Hierarchical Phrase-Based Translation with Jane 2. The Prague Bulletin of Mathematical Linguistics
  - This was shared work and severals experiments (first paper) have been run by myself:
    - [Peitz & Freitag$^+$ 11] Modeling Punctuation Prediction as Machine Translation
    - [Peitz & Freitag$^+$ 14] Better Punctuation Prediction with Hierarchical Phrase-Based Translation
  - I was responsible for the RWTH Aachen German→English evaluation during this project (machine translation as well as spoken language translation):
    - [Boudahmane & Buschbeck$^+$ 11] Advances on Spoken Language Translation in the Quaero Program
  - I implemented the MERT, a novel modified MIRA, and the Downhill Simplex algorithm:
    - [Stein & Vilar$^+$ 11] A Guide to Jane, an Open Source Hierarchical Translation Toolkit
  - I only had a minor part in the following papers. I helped writing the papers and discussing the ideas:
    - [Huck & Vilar$^+$ 13] A Performance Study of Cube Pruning for Large-Scale Hierarchical Machine Translation
    - [Huck & Peitz$^+$ 12a] Discriminative Reordering Extensions for Hierarchical Phrase-Based Machine Translation
- Machine Translation Evaluation Campaigns:
  - I developed the German preprocessing which is described in the paper. As part of this paper, I also presented new methods for language identification:
    - [Peitz & Wuebker$^+$ 14] The RWTH Aachen German-English Machine Translation System for WMT 2014
  - I ran all system combination experiments listed in the following papers:
    - [Wuebker & Peitz$^+$ 13a] The RWTH Aachen Machine Translation Systems for IWSLT 2013
    - [Peitz & Mansour$^+$ 13b] The RWTH Aachen Machine Translation System for WMT 2013
    - [Peitz & Mansour$^+$ 12] The RWTH Aachen Speech Recognition and Machine Translation System for IWSLT 2012
  - I ran half of the Japanese→English experiments:
    - [Feng & Schmidt$^+$ 13] The RWTH Aachen System for NTCIR-10 PatentMT
  - I ran the German→English experiments:
    - [Huck & Peitz$^+$ 12b] The RWTH Aachen Machine Translation System for WMT 2012
  - I ran half of the Spoken Language Translation experiments of the paper:
    - [Wuebker & Huck$^+$ 11] The RWTH Aachen Machine Translation System for IWSLT 2011

- I ran half of the Chinese→English experiments and extended the idea of reverse decoding:
  - [Feng & Schmidt$^+$ 11] The RWTH Aachen System for NTCIR-9 PatentMT
- I ran some of the German→English and English→German experiments:
  - [Huck & Wuebker$^+$ 11] The RWTH Aachen Machine Translation System for WMT 2011

2 Scientific Goals

In this thesis we pursue the following scientific goals:

- We introduce a novel system combination implementation which is integrated into RWTH's open source statistical machine translation toolkit Jane. On the most recent system combination shared task, we achieve improvements over the best submissions of the official evaluation. Moreover, we enhance the system combination pipeline with additional n-gram language models and lexical translation models which further improve the translation quality. We use the implementation which already performs best on the most recent system combination evaluation as baseline for all further investigations.

- We introduce a novel local system voting model trained by a neural network. This model enhanced the commonly used system voting features by taking the local word content and the combinatorial occurrences for its local system preference into account. This gives system combination the additional option to select words which have been only produced by one or few individual systems. System combinations including this novel feature outperform the baseline setup. We show that rarely seen words are now part of the consensus translation. Furthermore, this feature gives the community the opportunity to yield gains in system combination by adding hypotheses which generally perform worse, but are useful for few types of sentences.

- We introduce a novel alignment approach which aligns the individual system engines into a lattice from which the consensus translation can be extracted. The novel approach is based on the phrase information provided additional to the translations by different machine translation engines. We present results which further improve the previously presented high-quality system combination baseline. Additionally, we calculate the alignment error rates for different types of widely used alignment approaches and show that the proposed approach yields a lower alignment error rate.

- We investigate the decoding directions for the two commonly known statistical machine translation approaches phrase-based translation and hierarchical phrase-based translation. We reverse the word order of the source and/or target language and compare the translation results with the normal direction. Different to previous work, we also run the alignment training on partially or fully reversed corpora. Further, we present how to enhance translation quality by combining

machine translation systems trained with different word orders. Additionally to system combination, we also investigate combinations in an earlier step, e.g. alignment combination or phrases table combination.

- We present the recent evaluation results which were obtained with the system combination approach invented in this thesis. We compare our engines with the engines of world-leading research labs all over the world.
- All implementations are publicly available in RWTH's open source machine translation toolkit Jane. This gives the machine translation community the chance to use all the enhancements which we present in this thesis.

3 Preliminaries

In this chapter, we introduce some principles which are essential for a better understanding of the topics presented in this thesis. All below described methods have been invented and developed by different authors. We only give a brief overview of the methods which are used in this thesis and refer the reader to the cited papers for more details.

In the following chapters, we analyze two statistical machine translation engines, namely phrase-based statistical machine translation and hierarchical phrase-based statistical machine translation. Further, we put our focus on the approach of confusion network system combination to combine different machine translation approaches. All methods presented in this thesis rely on the idea of defining statistical models for the translation process which are automatically trained on large amount of data. Many concepts and algorithms used in statistical machine translation engines are adapted for confusion network system combination and the knowledge of the basic concepts are essential for the understanding of this thesis.

This chapter is organized as follows: We start with introducing some basic terminology in Section 3.1. We give a short introduction into statistical machine translation in Section 3.2. We present the log-linear model combination in Section 3.3. Section 3.4 describes the state-of-the-art error metrics in machine translation as well as the optimization algorithms for the free parameters of the log-linear framework. We conclude this chapter by giving a short introduction to phrase-based machine translation in Section 3.5 as well as to hierarchical phrase-based machine translation in Section 3.6.

3.1 Basic Terminology

We introduce the basic terminology which is used in the following chapters. A *hypothesis* is an automatic generated translation of a document in a foreign human language. A *reference* is a human translation which is used for measuring the quality of a hypothesis. A parallel collection of sentences given in two different languages is called a bilingual *corpus*. If a text is only available in one language, it is called *monolingual data*.

The usually huge amount of bilingual sentences is split into three different portions. The larges one is used for estimating and training various statistical models. A small portion called *development set* is used to optimize free parameters of the machine translation engines. A third portion called *test set* is used as a hidden data set which is kept untouched in the whole system build. The test set is used to

compare the translation quality of different translation setups and is needed for system development. Both development set and test set are normally of high quality and similar size.

The following terminology is used in the following chapters:

- A source sentence consisting of J words is denoted as: $f_1^J = f_1, \ldots, f_J$
- A target sentence consisting of I words is denoted as: $e_1^I = e_1, \ldots, e_I$
- A succession of n words will be denoted as an *n-gram*. For $n = 1, \ldots, 4$, n-grams are called *unigrams*, *bigrams*, *trigrams*, and *four-grams*.
- The predecessor words of a word in a sentence are called its *history*.

3.2 Statistical Machine Translation

In statistical machine translation, we calculate the a-posteriori probability of all possible target sentences to get a translation of a given source sentence. The target sentence that maximizes this probability is selected as the final hypothesis. In theory, we can formulate the decision process as:

$$f_1^J \rightarrow \hat{e}_1^{\hat{I}}\left(f_1^J\right) = \underset{e_1^I}{\operatorname{argmax}} \left\{ Pr(e_1^I | f_1^J) \right\} . \tag{3.1}$$

In Equation 3.1, we apply Bayes' theorem to split the probability $Pr(e_1^I | f_1^J)$ into two further probability distributions:

$$f_1^J \rightarrow \hat{e}_1^{\hat{I}}\left(f_1^J\right) = \underset{e_1^I}{\operatorname{argmax}} \left\{ \frac{Pr(e_1^I) \cdot Pr(f_1^J | e_1^I)}{Pr(f_1^J)} \right\} \tag{3.2}$$

$$= \underset{e_1^I}{\operatorname{argmax}} \left\{ Pr(e_1^I) \cdot Pr(f_1^J | e_1^I) \right\} \tag{3.3}$$

The denominator of Equation 3.2 can be omitted as it is constant and does not affect the maximum. Finally, the formula consists of two models: the *language model* $Pr(e_1^I)$ and the *translation model* $Pr(f_1^J | e_1^I)$. We address both models below.

3.2.1 Language Model

A language model is an important feature in statistical machine translation and assigns each target sentence e_1^I a probability $Pr(e_1^I)$. Roughly speaking, it measures the fluency and grammatical correctness of the sentence without taking the source sentence into account. The language model is trained on huge amounts of monolingual data. In theory, we need to calculate the probability for each word e_i based on its full history:

$$Pr(e_1^I) = \prod_{i=1}^{I} Pr(e_i | e_1^{i-1}) . \tag{3.4}$$

Typically, we have to restrict the language model in the way that it only considers a constant history size. This limitation has not only the benefit of lower computational effort, it is also more likely that

e.g. an four-gram has been seen during training than the complete sentence. A correct sentence which has not been seen during training is assigned a low probability. A language model considering only $n-1$ words $h_i = e_{i-n+1}^{i-1}$, can be effective trained and gives a good approximation of commonly used n-grams in a language. In this thesis n is either 3, 4, or 6.

The quality of a language model can be computed by the *perplexity* on a development set. The perplexity measures the language model score of the target side of the development set. As the development set should be in fluency and grammatical correct language, the language model probability of the document should be high for a good language model. The calculation of the perplexity is defined as:

$$PP = \Pr(e_1^I)^{-\frac{1}{I}} \tag{3.5}$$

3.2.2 Translation Model and Alignment

Contrary to the language model, the translation model $Pr(f_1^J|e_1^I)$ of Equation 3.3 considers the source sentence. The translation model assigns a probability to one sentence f_1^J of being the translation of another sentence e_1^I. The calculation of the translation model relies on a *word alignment* which aligns each target word to a word of the source sentence. To model words that might not be translated into the other language at all, we also introduce the *empty word*. The alignment model is trained among others on statistical occurrences of word pairs on the bilingual corpus. Without loss of generality, we can now reformulate the translation model as:

$$\begin{aligned} Pr(f_1^J|e_1^I) &= \sum_{\mathscr{A}} Pr(f_1^J, \mathscr{A}|e_1^I) && (3.6)\\ Pr(f_1^J, \mathscr{A}|e_1^I) &= Pr(J|e_1^I) \cdot Pr(f_1^J, \mathscr{A}|J, e_1^I) && (3.7)\\ &= Pr(J|e_1^I) \cdot Pr(\mathscr{A}|J, e_1^I) \cdot Pr(f_1^J|\mathscr{A}, J, e_1^I)\,. && (3.8) \end{aligned}$$

We split the formula into subproblems and call them the *length model* $Pr(J|e_1^I)$, the *alignment model* $Pr(\mathscr{A}|J, e_1^I)$, and the *lexicon model* $Pr(f_1^J|\mathscr{A}, J, e_1^I)$. In one of the fundamental publications in statistical machine translation, [Brown & Della Pietra$^+$ 93] introduced the *IBM models* which model our subproblems. The IBM models are designed to build upon each other, i.e. IBM Model 1 is the simplest model and its formula ignores the word positions in the sentence. IBM Model 1 is a *zero-order* model, as it ignores the surrounding words. Nevertheless, this simple model is used as initialization for higher models. The estimates can be improved in IBM Model 2, passed on to IBM Model 3 and so on. We continue to briefly introduce these models, but refer the reader to the cited paper for further understanding.

In IBM Model 1 we make the assumption that each source position is equally likely to be aligned to each target position. The word order in both source and target sentence does not affect the probabilities. In IBM Model 2 we model in addition that the probability of an alignment point depends on the positions it connects and on the lengths of source and target sentence. As a result, for IBM Model 2 the probabilities highly depend on the word order of the source and target sentences. In practice IBM Model 2 is not modelled with the word positions themselves, but with the absolute difference between the source and target word position. This results to the fact, that the dominating factor in the alignment model of IBM Model 2 is the diagonal line of an alignment chart.

IBM Model 3 is extended by a *word fertility* ϕ_j model. This model allows for certain words in one language to produce more than one word in the other. In this thesis, we employ in addition to the IBM Models an extension of IBM Model 2 that is based on a Hidden Markov Models (HMM). The

HMM alignment model [Vogel & Ney$^+$ 96] is a first-order model, because it also takes the preceding alignment position into account.

Based on their definitions, the IBM Models lead to different alignments when switching the source and the target sides. In this thesis, we train alignments for both language directions and merge both alignments with symmetrization heuristics (e.g. grow-diag-final, iu, intersection, or union as described in [Och & Ney 03]).

In all of our experiments, we run the following alignment training setup and use the result of the previous model as initialization for the next model:

- IBM Model 1 (5 iterations)
- HMM (5 iterations)
- IBM Model 4 (5 iterations)

3.3 Log-Linear Model

We mathematically defined the automatic translation process in Equation 3.3 and introduced the two major components language model and translation model. In practice, we want to add more statistical models into the translation process to increase the translation quality. In 2002, [Och & Ney 02] extended the mathematical foundation of statistical machine translation with a *log-linear model*, which models the a-posteriori probability:

$$p(e_1^I|f_1^J) = \frac{\exp\left(\sum_{m=1}^M \lambda_m h_m(f_1^J, e_1^I)\right)}{\sum_{\tilde{e}_1^I} \exp\left(\sum_{m=1}^M \lambda_m h_m(f_1^J, \tilde{e}_1^I)\right)}. \tag{3.9}$$

The log-linear model contains a set of M different statistical models $h_m(f_1^J, e_1^I)$. Each of them is assigned one scaling factor λ_m which determines the impact of each model. It is easy to add several new models into the log-linear framework which can be based on totally different approaches. The definition of the log-linear model ensures that we always stay mathematically correct.

The denominator in Equation 3.9 can be omitted as it does not rely on the translation e_1^I and does not change the decision of $\hat{e}_1^{\hat{I}}$:

$$\begin{aligned}
f_1^J \rightarrow \hat{e}_1^{\hat{I}}\left(f_1^J\right) &= \operatorname*{argmax}_{e_1^I} \left\{ p(e_1^I|f_1^J) \right\} && (3.10)\\
&= \operatorname*{argmax}_{e_1^I} \left\{ \frac{\exp\left(\sum_{m=1}^M \lambda_m h_m(f_1^J, e_1^I)\right)}{\sum_{\tilde{e}_1^I} \exp\left(\sum_{m=1}^M \lambda_m h_m(f_1^J, \tilde{e}_1^I)\right)} \right\} && (3.11)\\
&= \operatorname*{argmax}_{e_1^I} \left\{ \frac{\sum_{m=1}^M \lambda_m h_m(f_1^J, e_1^I)}{\sum_{\tilde{e}_1^I} \left(\sum_{m=1}^M \lambda_m h_m(f_1^J, \tilde{e}_1^I)\right)} \right\} && (3.12)\\
&= \operatorname*{argmax}_{e_1^I} \left\{ \sum_{m=1}^M \lambda_m h_m(f_1^J, e_1^I) \right\} && (3.13)
\end{aligned}$$

Two possible models of the log-linear approach are the previously introduced language model and translation model. Besides that several more complex models are introduced as part of this thesis. The model choice always depends on several components as language pair, translation task, or system architecture. The general approach is shown in Figure 3.1.

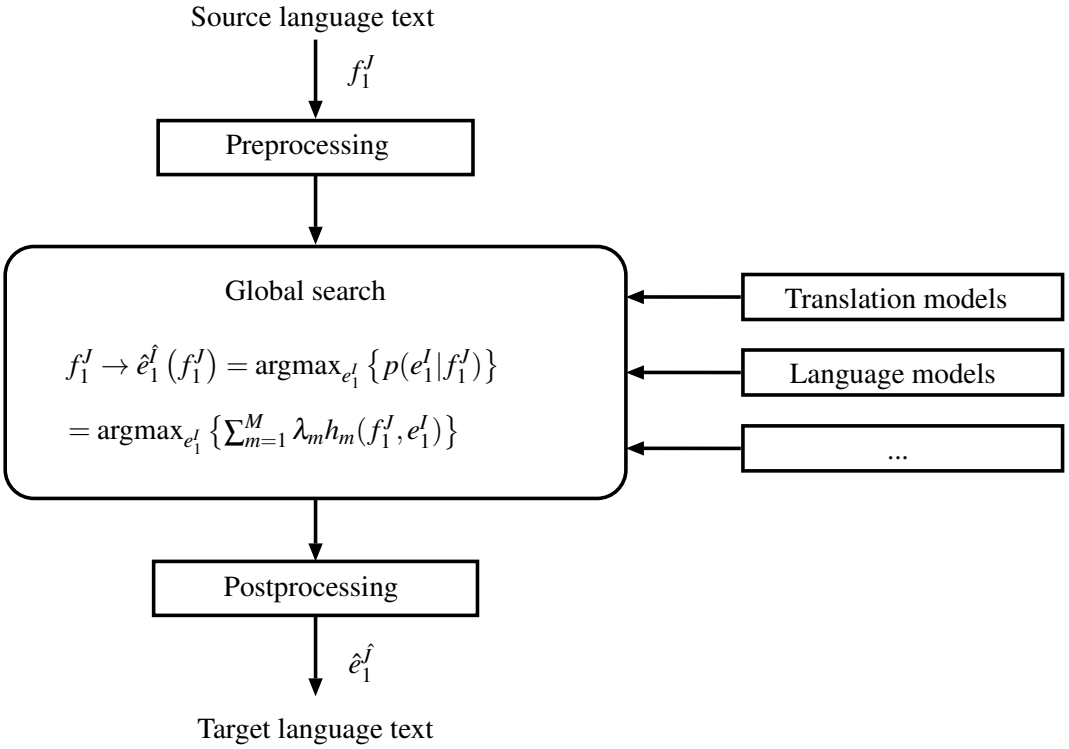

Figure 3.1: Illustration of the log-linear translation model. An arbitrary number of models can be used in this approach.

3.4 Optimization

Each statistical model in Equation 3.13 is assigned one scaling factor λ_m which defines the impact of each model. In this section, we describe an algorithm which determines the best set of scaling factors for a given translation setup on a held-out development set. Before coming to the optimization method, we first need to introduce several error metrics which determine the quality of a hypothesis. As human evaluation is costly and time consuming, we introduce various automatic error metrics whose scores are calculated based on a human generated reference translation.

3.4.1 Error Metrics and Scores

As part of this thesis, we use the following error metrics for evaluations:

WER: The Word Error Rate (WER) is defined as the Levenshtein distance [Levenshtein 66] of words between a hypothesis and a reference translation. The WER score is the minimum number of insertions, substitutions, and deletions to modify the hypothesis to match the reference translation divided by the reference length. Nowadays, the WER is the state-of-the-art error metric in automatic speech recognition and is not used for machine translation evaluation anymore.

TER: Contrary to automatic speech recognition, in machine translation correct translations can have various different word orders and we need to capture word reorderings in the evaluation. The Translation Edit Rate (TER) [Snover & Dorr$^+$ 06] is an extension of the WER. It tries to capture word reorderings and allows in addition to the operations of WER to shift blocks of words to match the reference translation.

BLEU: The Bilingual Evaluation Understudy (BLEU) [Papineni & Roukos$^+$ 02] score is defined as the n-gram precision of the hypothesis and its reference. Precision metrics usually prefer shorter sentences. To overcome this problem, BLEU includes a brevity penalty which penalize to short hypotheses.

In this thesis, we throughout use the BLEU and the TER error metrics for evaluating and testing the novel methods.

3.4.2 Minimum Error Rate Training

Based on the previously introduced error metrics, we present the *Minimum Error Rate Training (MERT)* which finds the best scaling factors for the log-linear model that maximize an chosen error metric. First, we give MERT a set of n-best possible translations as an approximation of the full search space (called n-best list). MERT then optimizes one scaling factor at a time, in a random order and uses the fact that when changing one scaling factor λ_k, the translation score $f(\lambda) = \sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J)$ of one hypothesis is a linear function of one variable λ_k:

$$f(\lambda_k) = \lambda_k h_k(e_1^I, f_1^J) + \sum_{m=1, m\neq k}^{M} \lambda_m h_m(e_1^I, f_1^J) \tag{3.14}$$

The optimization based on n-best lists is only an approximation of the whole search space. Therefore, we run MERT several iterations and produce in each iteration the n-best translations regarding to the actual scaling factors. We merge the newly generated n-best list with the previous ones to get a larger approximation of the search space. The optimization process ends when no unseen hypothesis has been generated by the current scaling factors (usually after 6-8 iterations) which means that the approximated search space did not change.

Algorithm 1: n-best List Optimization

```
initialization: create n-best list with initial configuration;
while new hypotheses have been produced do
    find best scaling factors with MERT;
    generate new n-best lists with optimized scaling factors;
    merge with old n-best lists;
```

3.5 Phrase-Based Machine Translation

Going from theory to practice, it is not feasible to score all possible translations. We have to limit the search space and only consider a reasonable amount of translation options. There are a variety of translation approaches which try to generate and test only the best possible translation options. One of the most famous approaches is phrase-based translation (PBT). In state-of-the-art phrase-based machine translation systems, the word alignments are usually modeled implicitly through bilingual phrases. The basic idea of phrase-based translation is to first segment the source language sentence into phrases, then translate each phrase, and finally compose the target sentence of these phrase translations. PBT is motivated by the fact that the context is important in translation. The corresponding phrase segmentation of an alignment example is depicted in Figure 3.2. Phrase pairs are represented as boxes.

For all phrase-based experiments, we apply the open source toolkit `Jane`. The phrase-based decoding algorithm in `Jane` is a source cardinality synchronous search (SCSS) procedure and applies

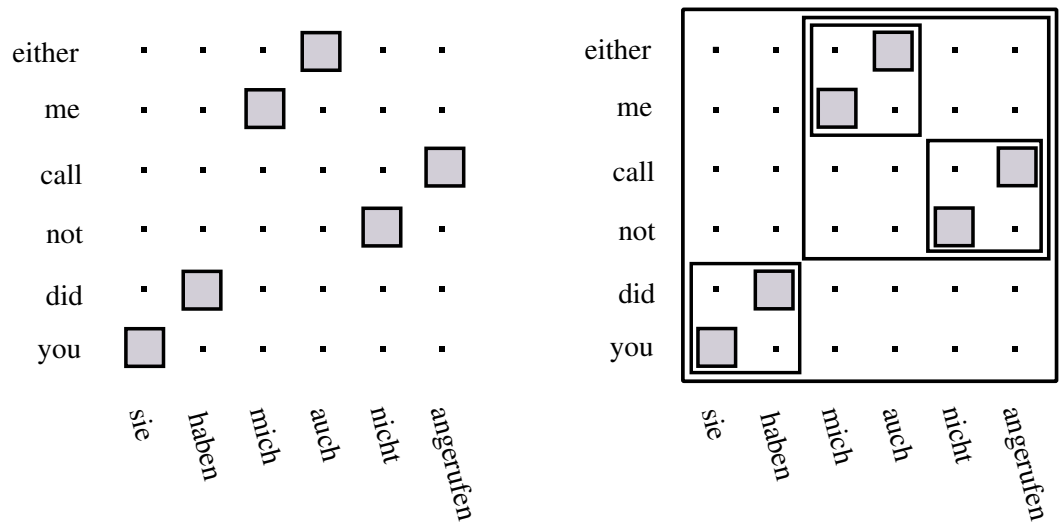

Figure 3.2: Left hand side: word-to-word alignment. Right hand side: corresponding phrases.

separate pruning to lexical and coverage hypotheses similar to [Zens & Ney 08]. The distinction between lexical and coverage hypotheses has been shown to have a significant positive effect on the scalability of the algorithm. For efficient decoding, language model look-ahead [Wuebker & Ney$^+$ 12] can be applied. The models we used during decoding are:

- Lexical smoothing probabilities in both translation directions
- Phrase translation probabilities in both translation directions
- Word penalty
- Phrase penalty
- Enhanced low frequency feature [Chen & Kuhn$^+$ 11]
- Jump distance limit [Zens & Ney$^+$ 04]
- n-gram language model
- Word class n-gram language model [Wuebker & Peitz$^+$ 13b]
- Hierarchical reordering model [Galley & Manning 08]

3.6 Hierarchical Phrase-Based Machine Translation

An extension of the phrase-based translation approach is the hierarchical phrase-based (HPBT) [Chiang 05] approach. In hierarchical phrase-based translation, a weighted synchronous context-free grammar is induced from parallel text. In addition to contiguous lexical phrases, hierarchical phrases with up to two gaps are extracted. The search is carried out with a parsing-based procedure. We utilize the cube pruning algorithm [Chiang 07] for decoding. The standard models integrated into the `Jane` HPBT systems are:

- Lexical smoothing probabilities in both translation directions

- Phrase translation probabilities in both translation directions
- Word penalty
- Phrase penalty
- Enhanced low frequency feature [Chen & Kuhn$^+$ 11]
- Binary feature marking hierarchical phrases
- Glue rule
- Paste rule
- *n*-gram language model
- Word class *n*-gram language model [Wuebker & Peitz$^+$ 13b]
- Hierarchical reordering model [Galley & Manning 08]

4

Jane: Open Source MT System Combination

The combination of outputs from multiple machine translation (MT) systems has been successfully applied in state-of-the-art machine translation evaluations for several years. Current machine translation systems are based on different paradigms, such as bilingual phrases, hierarchical phrases, hand-crafted rules, syntactically derived rules or neural networks. System combination is a reliable method to combine the benefits of different machine translation systems into a single high quality translation output. System combination relies on the concept of majority voting and on the assumption that different machine translation systems produce different errors at different positions, but the majority agrees on a correct translation at each position. In the last years, there has been extensive research in the field of system combination (syscomb).

System combination methods proposed in the literature can be roughly divided into three categories: hypothesis selection, re-decoding, and confusion network decoding. In hypothesis selection, the challenge is to select sentence-wise one of the different individual system outputs as new output. In contrast to the following two approaches, hypothesis selection is unable to generate new sentences which are different from any given individual system output. In re-decoding, a new phrase table based on the given individual system outputs is generated. With the new phrase table and some new similarity models, this approach uses a general statistical machine translation decoder to re-decode the source sentence. The third and most successful approach is confusion network decoding in which the individual system outputs are word-to-word aligned. From the calculated alignment information, a confusion network is built from which the system combination output is determined using majority voting and additional models. In recent years, confusion network system combination emerged as one of the most successfully applied approaches in combining translation outputs generated by different machine translation engines.

As part of this thesis, we invented a novel system combination framework which has been released as part of RWTH Aachen's machine translation toolkit `Jane`[1]. The system combination framework has been applied successfully for joining the outputs of different individual machine translation engines within large-scale projects like Quaero [Freitag & Peitz$^+$ 12, Peitz & Mansour$^+$ 13a], EU-BRIDGE [Freitag & Peitz$^+$ 13, Freitag & Peitz$^+$ 14, Freitag & Wuebker$^+$ 14], and DARPA BOLT[2]. We will show

[1] `Jane` is publicly available under an open source non-commercial license and can be downloaded from `http://www.hltpr.rwth-aachen.de/jane/`.

[2] RWTH Aachen was part of the IBM team in the DARPA Broad Operational Language Translation (BOLT) program.

that our novel system combination implementation, which has been implemented as part of this thesis, performs up to 0.7 points better in both TER and BLEU than the best evaluation system combination submissions on all investigated WMT 2011 system combination shared tasks. Moreover, we reach the best performance on all tasks which is remarkable as each task has been won by a different group (and consequently by a different approach). Compared to a single group or approach, the improvements are even much larger.

4.1 Introduction

We present a new system combination framework which relies on the concept of confusion network decoding. The original idea is based on the ROVER approach of [Fiscus 97] for combining speech recognition hypotheses in a confusion network. ROVER combines hypothesized word outputs of multiple recognition systems and selects the best scoring word sequence. The approach can be divided into three major steps: first, we need to generate pairwise alignments between the words of the different system outputs. Second, a confusion network is built based on the previous calculated alignment information. Finally, all arcs in the network get assigned model scores to evaluate the different translation options. The highest scoring path is selected as the combined system output.

In contrast to machine translation system combination, the speech recognition hypotheses share the same word order and can easily be aligned. To generate a confusion network in machine translation system combination, pairwise word alignments of the individual machine translation hypotheses have to be learned using an alignment algorithm which captures word reorderings. The confusion network is then scored by different models and the most probable translation can be extracted as an improved translation output. An overview of the confusion network approach is illustrated in Figure 4.1. Confusion network system combination includes the following main steps:

- Generate word-to-word alignments between all pairs of input hypotheses
- Build a confusion network based on the previous calculated alignment information
- Rescore the network with several models
- Extract the path with the highest model scores from the confusion network

For the confusion network algorithm we only need the first best translation from each of the different machine translation engines, without any additional information. We integrate hypotheses language model, word penalty, system voting models, *n*-gram language models and IBM-1 lexicon models [Brown & Della Pietra$^+$ 93] in our framework. The last two are trained on additional training corpora, which might be at hand.

We evaluate the `Jane` system combination framework on the latest official *Workshop on Statistical Machine Translation* (WMT) system combination shared task [Callison-Burch & Koehn$^+$ 11]. Many state-of-the-art machine translation system combination toolkits have been evaluated on this task, which allows us to directly compare the results obtained with our novel `Jane` system combination framework with the best known results obtained with other toolkits. Further, results are presented on the DARPA BOLT Arabic→English as well as on the DARPA BOLT Chinese→English tasks for which we were responsible to combine up to 8 different translation engines for the Delphi IBM team. The main parts of this chapter has been published and described in [Freitag & Huck$^+$ 14].

The chapter is structured as follows: We commence by giving an outline of related work (Section 4.2). In Section 4.3, we describe the techniques that are implemented in the `Jane` machine translation system combination framework. Experimental results are presented and analyzed in Section 4.4. We conclude this chapter in Section 4.5.

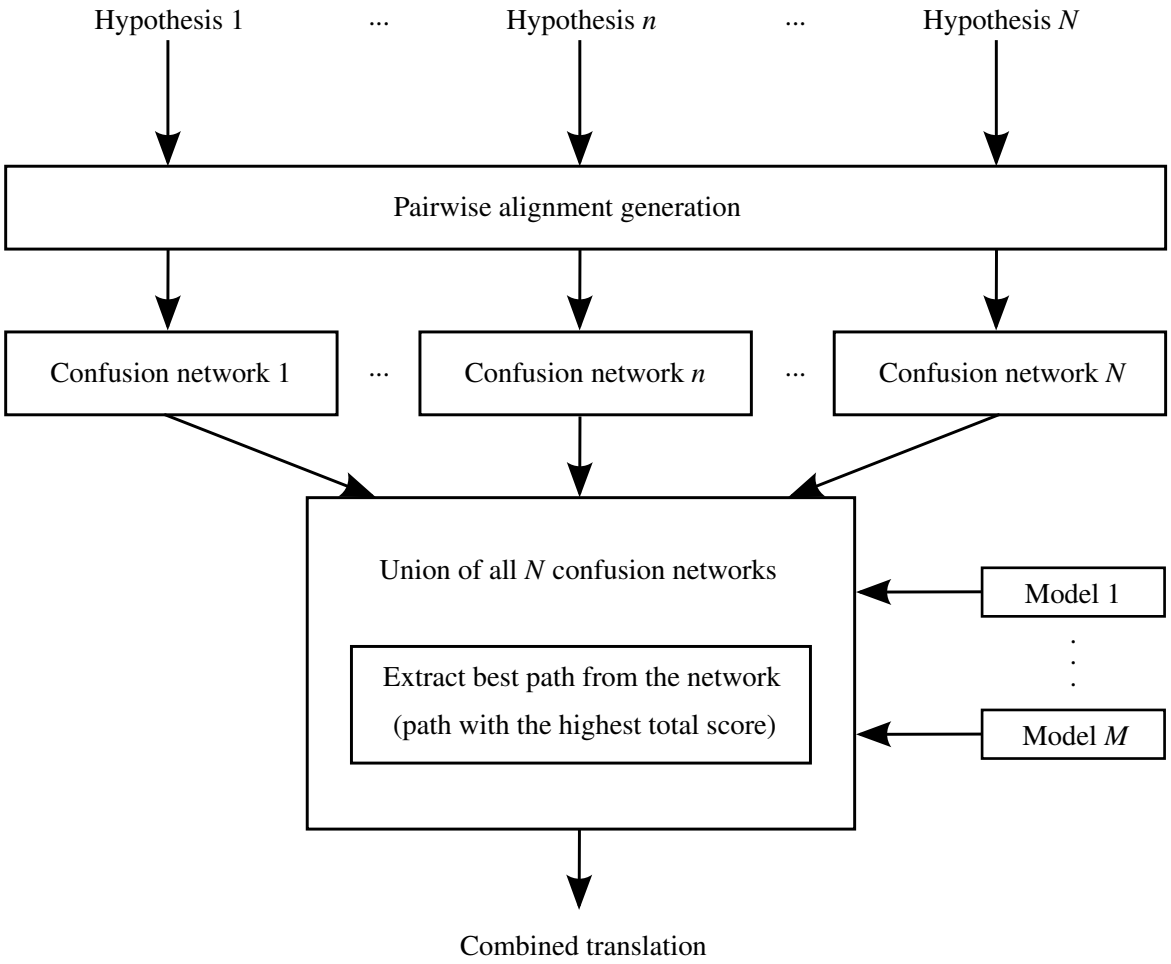

Figure 4.1: Overview of confusion network system combination constructed by N input hypotheses. First, pairwise alignments between all input hypotheses are calculated. Based on the alignment information, N different confusion networks are constructed. All confusion networks are scored by M models. The final translation is extracted from the highest scoring path of all confusion networks.

4.2 Related Work

System combination methods proposed in the literature can be roughly divided into three main approaches: hypothesis selection, re-decoding, and confusion network decoding. All approaches take as input a set of previously generated translations and either select one of the input translation (hypothesis selection) or combine all translations into a new output (re-decoding or confusion network decoding). In the following, we present the most important papers related to these three approaches. As a minor remark: in this thesis, we focus on confusion network system combination.

Hypothesis selection: Hypothesis selection selects one translation from a set of automatically generated candidate translations as new output. The important task in hypothesis selection is to find valuable models to choose the best hypothesis from the different system outputs.

[Nomoto 04] proposes to select the final translation via a combination of one language model and one translation model score. In addition, the author gives his system the option to choose the language model out of a pool of pretrained language models on a sentence level basis. Experiments are done using corpora from three different domains. The author

concludes that the voted language model extension lead to an improvement compared to a sentence selection with one fixed language model.

[Rosti & Ayan$^+$ 07] propose to re-rank merged *n*-best lists produced by different translation engines. A confidence score for each system is assigned to each unique hypothesis. The confidence scores for each hypothesis are used to produce a single score which, combined with a 5-gram language model score, determines a new ranking of the hypotheses.

[Hildebrand & Vogel 08] combine *n*-best lists from all input systems without the internal translation system scores. Besides the *n*-best list, no further information from the input systems is needed, which makes it possible to also include non-statistical translation systems in the combination. The authors use a language model and various *n*-gram agreement models to assign reliable scores to the different hypotheses.

Re-decoding: The second category is called re-decoding. This approach uses the internal phrase segmentations of the individual systems to generate a new phrase table. In addition to the standard models, the authors add new models to each phrase table entry. As the final step, this approach uses a traditional machine translation engine to re-decode the source sentences with the given phrases.

[Frederking & Nirenburg 94] use the translation segments of the individual system engines, and put the resulting output into a shared chart-like data structure. Next, all the partial translations are given an internal quality score. A chart-walk algorithm is used to find the best combination of the partial translations.

[Rosti & Ayan$^+$ 07] derive a new phrase translation table from the phrases used by the individual system engines to generate its translations. The phrase translation scores are based on the level of agreement between the system outputs and sentence posterior estimates. A standard phrase-based decoder is used to produce the final combination output. Additional to the presented re-decoding approach, the authors compare their approach to sentence selection and confusion network system combination approaches.

[He & Toutanova 09] propose an joint optimization approach for word-level combination of multiple machine translation hypotheses. Decisions on word alignment between hypotheses, word ordering, and the lexical choice of the final output are made jointly according to a set of models in the decoding process. Decoding is based on a beam search algorithm similar to the phrase-based machine translation decoder.

Confusion network decoding: In confusion network decoding, pairwise alignments between all system outputs are generated. From the calculated alignment information, a confusion network is built from which the system combination output is determined using majority voting and additional models. The hypothesis alignment algorithm is a crucial part of building the confusion network and many alternatives have been proposed in the literature.

[Bangalore & Bordel$^+$ 01] use a multiple string alignment (MSA) algorithm to identify the unit of consensus and applied a posterior language model to extract the consensus translations. In contrast to the following approaches, MSA is unable to capture word reorderings.

[Matusov & Ueffing$^+$ 06] produce pairwise word alignments with the statistical alignment algorithm toolkit GIZA++ that explicitly models word reordering. The context of a whole document of translations rather than a single sentence is taken into account to produce the alignments.

[Sim & Byrne$^+$ 07] construct a consensus network by using TER [Snover & Dorr$^+$ 06] alignments. Minimum bayes risk decoding is applied to obtain a primary hypothesis to which all other hypotheses are aligned.

[Rosti & Ayan$^+$ 07] extend the TER alignment approach and introduce an incremental TER alignment which aligns one system at a time to all previously aligned hypotheses.

[Karakos & Eisner$^+$ 08] use the inversion transduction grammar (ITG) formalism [Wu 97] and treat the alignment problem as a problem of bilingual parsing to generate the pairwise alignments.

[He & Yang$^+$ 08] propose an indirect hidden markov model (IHMM) alignment approach to address the synonym matching and word ordering issues in hypothesis alignment. Unlike traditional HMMs whose parameters are trained via maximum likelihood estimation (MLE), the parameters of the IHMM are estimated indirectly from a variety of sources including word semantic similarity, word surface similarity, and a distance-based distortion penalty.

[Barrault 10] describes a push-the-button MT system combination toolkit. The combination is based on the creation of a lattice made on several confusion networks connected together. This lattice is then decoded with a token-pass decoder to provide the best and/or n-best outputs. Each confusion network is built using a modified version of the TERP tool.

[Heafield & Lavie 10] use the METEOR toolkit to calculate pairwise alignments between the hypotheses. The METEOR automatic evaluation metric scores machine translation hypotheses by aligning them to one or more reference translations. Alignments are based on exact, stem, synonym, and paraphrase matches between words and phrases.

This chapter presents a novel open-source confusion network decoding implementation, which is a mix of the previously presented confusion network approaches. The toolkit is able to apply the GIZA++ as well as the METEOR alignment algorithms similar to [Matusov & Ueffing$^+$ 06] and [Heafield & Lavie 10]. The idea of incremental alignment ([Rosti & Ayan$^+$ 07]) is applied for both alignment algorithms. Instead of using one primary hypothesis, which is responsible for the word order, we build N confusion networks each having one of the N input hypotheses as primary hypothesis. The final network is a union of the N confusion networks similar to [Matusov & Ueffing$^+$ 06]. Instead of using the beam search algorithm, we use the shortest path algorithm to extract the combined translations. Furthermore, we have the advantage to use both the optimization tools and the models of the phrase-based and hierarchical phrase-based translation engines as they have been previously implemented into Jane. The choice of which alignment algorithm, which confusion network structure and which optimization algorithm to use is a result of many experiments. All confusion network approaches have been fully reimplemented and we used the best performing combination of components for our Jane setup. Different to all previous work, we enhanced our setups with IBM-1 probabilities tables to keep connection to the source sentence.

4.3 The Jane MT System Combination Framework

In this section we describe the techniques for machine translation system combination which we implemented in the Jane toolkit[3]. We first address the generation of a confusion network from the input translations. For that we need a pairwise alignment between all input hypotheses. We then present word reordering mechanisms and the models which can be applied for system combination using Jane. The decoding step basically involves determining the shortest path through the confusion network based on previously calculated model scores.

[3]Practical usage aspects are explained in the manual: http://www.hltpr.rwth-aachen.de/jane/manual.pdf

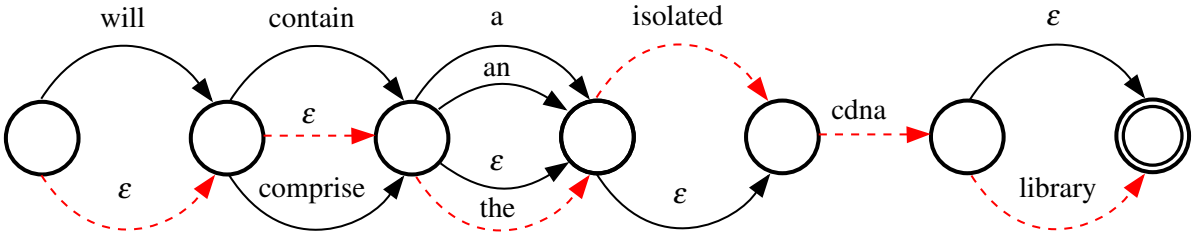

Figure 4.2: Example confusion network. ε denotes the empty word which gives the decoder the option to skip words. The red dashed arcs highlight the shortest path which will be later determined by different statistical models.

4.3.1 Confusion Network

A confusion network is a linear graph in which all possible paths visit all nodes. This is the reason why a confusion network is often called sausage. For system combination, the confusion network represents all different combined translations we can generate from the set of provided input hypotheses. A word alignment between all pairs of input hypotheses is required for generating a confusion network. For convenience, we first select one of the input hypotheses as the primary hypothesis. The primary hypothesis then determines the word order and all remaining hypotheses are word-to-word aligned to the word order of the primary hypothesis. Figure 4.2 depicts an example of a confusion network built by N=6 input hypotheses. Every arc represents a choice between up to N different words.

To generate a meaningful confusion network, we should adopt an alignment that only allows to switch between words which are synonyms, misspellings, morphological variants or on a higher level paraphrases of the words from the primary hypothesis. In this thesis, we investigate two different alignment algorithms. First, we use METEOR alignments. METEOR [Denkowski & Lavie 11] was originally designed to reorder a translation for scoring and has a high precision as it only relies on exact word matches, synonyms, stems, or paraphrases. Second, we use GIZA++ which trains an alignment in analogy to the alignment procedure in statistical machine translation. GIZA++ is a statistical machine translation toolkit that is used to train IBM Models 1-5 and a HMM word alignment model. GIZA++ uses a statistical algorithm and can also produce alignment points that do not match the above mentioned criteria. It has been proven to work well for aligning different system outputs [Matusov & Ueffing$^+$ 06].

An example output of 6 different machine translation systems is given in Figure 4.3. Without loss of generality, we choose the first system to be the primary system that is responsible for the word order (the other outputs are reordered to match the word order of the primary one). We present the selection algorithm for the primary system later. We then calculate a pairwise alignment for each non-primary input hypotheses with the primary hypothesis. The result is illustrated in Figure 4.4. The primary hypothesis "contain isolated cdna library" determines the word order. An entry "a|b" means that word "a" from a secondary hypothesis has been aligned to word "b" from the primary one. "ε" is the empty word and thus an entry "ε|b" means that no word is aligned to the primary hypothesis word "b". "a|ε " means that the word "a" has not been aligned to any word from the primary hypothesis. The empty words will be discussed in Section 4.3.2.

The METEOR database only contains synonyms and paraphrases. Punctuation marks like "!" and "?" are not aligned to each other. For our purposes, we augment the METEOR paraphrase table with entries like ".|!", ".|?", or "the|a" to give the decoder the possibility to choose between these options. A complete list of all added entries is given in Figure 4.5.

Based on the alignment information, we build the confusion network. This is basically a mapping of the word-to-word alignment into the network structure. In Figure 4.2 the confusion network gen-

primary system	contain isolated cdna library
secondary systems	compromise the isolated cdna
	will contain a isolated cdna library
	an isolated cdna library
	the cdna library
	the cdna

Figure 4.3: Example of 6 different system outputs. Without loss of generality, we choose the first system as primary system.

Secondary	Alignment to primary hypothesis 1					
Hypothesis 2	ε|ε	comprise|contain	the|ε	isolated|isolated	cdna|cdna	ε|library
Hypothesis 3	will|ε	contain|contain	a|ε	isolated|isolated	cdna|cdna	library|library
Hypothesis 4	ε|ε	ε|contain	an|ε	isolated|isolated	cdna|cdna	library|library
Hypothesis 5	ε|ε	ε|contain	the|ε	ε|isolated	cdna|cdna	library|library
Hypothesis 6	ε|ε	ε|contain	the|ε	ε|isolated	cdna|cdna	ε|library

Figure 4.4: Alignment result after running METEOR. ε denotes the empty word. The primary hypothesis is "contain isolated cdna library". An entry "a|b" means that word "a" from a secondary hypothesis has been aligned to word "b" from the primary one.

a	$\Leftrightarrow$	the
.	$\Leftrightarrow$	!
.	$\Leftrightarrow$	?
.	$\Leftrightarrow$	,
,	$\Leftrightarrow$	!
,	$\Leftrightarrow$	?
,	$\Leftrightarrow$	;
?	$\Leftrightarrow$	!

Figure 4.5: We add additional entries to the synonym table of the METEOR toolkit to tackle pairs of words which are no synonyms but should be aligned in system combination.

erated by the alignment information of Figure 4.4 is illustrated. Now, we are able to not only extract the original primary hypothesis from the confusion network but also switch words from the primary hypothesis to words from any secondary hypothesis (also the empty word) or insert words or sequences of words.

The most straightforward way to obtain a combined hypothesis from a confusion network is to extract it via majority voting. For instance, in Figure 4.6, "the" has been seen three times, but the translation options "a" and "an" have each been seen only once. By means of a straight majority vote we extract "the". Nevertheless, the different input hypotheses (from which the confusion network has been built) can be of different value for the final word choices and an unweighted majority vote could lead to not optimal result. As a consequence, we assign a system weight to each input hypothesis and

Hypothesis 1	ε	contain	ε	isolated	cdna	library
Hypothesis 2	ε	compromise	the	isolated	cdna	ε
Hypothesis 3	will	contain	a	isolated	cdna	library
Hypothesis 4	ε	ε	an	isolated	cdna	library
Hypothesis 5	ε	ε	the	ε	cdna	library
Hypothesis 6	ε	ε	the	ε	cdna	ε
Majority vote	ε	ε	the	isolated	cdna	library

Figure 4.6: Majority vote on the previously generated alignment. The last line is the system combination output which prefers the words which has been produced most. E.g. "library" is part of the system combination output as it appears 4 times whereas "ε" only appears twice.

perform a weighted majority vote on the network. The weights are obtained by parameter optimization on a development test. The optimization algorithm will be presented in Section 4.3.3.

Up to now, we stuck to one primary system hypothesis which defines the word order of the system combination output. We want to give the decoder the option to choose between all word orders given by the different input hypotheses. Consequently, we build N different confusion networks, each having a different system as primary system. The final network is a union of all N confusion networks. As a result of the union, all possible paths do not visit all nodes anymore and thus the resulting network is no confusion network anymore. An example network is illustrated in Figure 4.7.

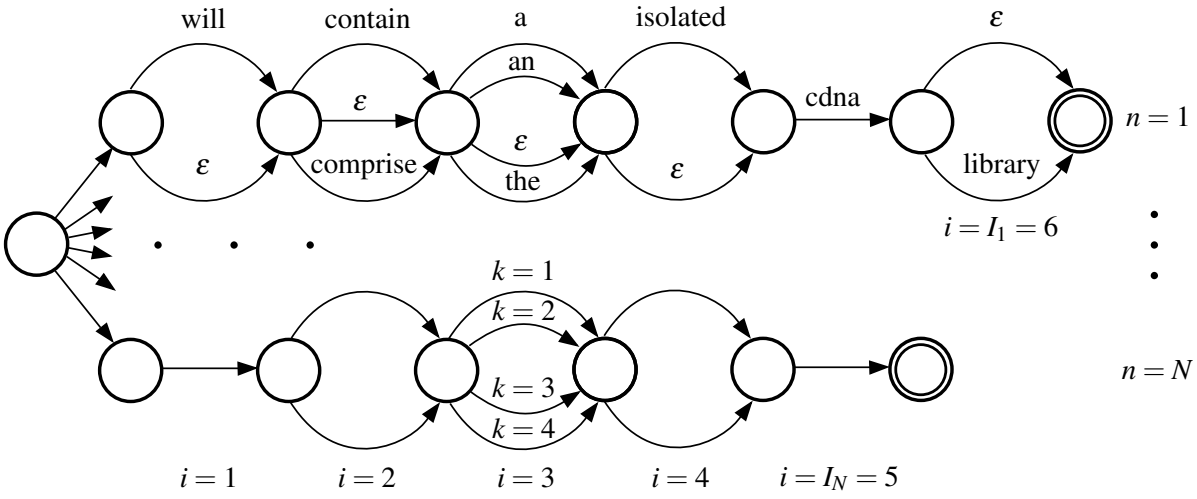

Figure 4.7: Union of N different confusion networks, each constructed with a different primary system. The confusion networks can be of different length I_n as the number of ε tokens depends on the primary system. Between two nodes, the best arc k of up to N different words has to be determined.

4.3.2 Unaligned Words (Governed Insertion)

Many words from non-primary hypotheses can be unaligned as they have no connection to any words of the primary hypothesis. Usually, we put these words next to the previous successfully aligned word and insert an epsilon arc for the primary system and all previous inserted secondary hypotheses.

However, words from different secondary systems could be related to each other. To account for these relations and to give the words from the secondary hypotheses a higher chance to be present in the combined output, we introduce some simple word reordering mechanisms.

We rank the hypotheses according to a language model trained on all input hypotheses. We initialize the confusion network with the sentence from the primary system. During the generation of the confusion network we align the hypotheses consecutively into the confusion network via the following procedure:

1. If a word w_i from hypothesis A has a relation to a word v_j of the primary hypothesis, we insert it as a new translation alternative to v_j.
2. If w_i has no relation to the primary, but to a word u_k from a secondary hypothesis in the confusion network, we insert w_i as a new translation alternative to u_k.
3. Otherwise we insert w_i behind the previous inserted word w_{i-1} of hypothesis A. The new position gets an epsilon arc for the primary and all unrelated secondary systems.

For experiments without governed insertion, we skip step 2 during the confusion network generation and no relations between secondary hypotheses are taken into account.

4.3.3 Models

Once the network has been built, we score the arcs of the network with different models. We call the set of the following models the standard models as they are used in all experiments in this thesis:

N binary system voting models: For each word the voting model for system n ($1 \leq n \leq N$) is 1 iff the word is from system n, otherwise 0. This model assigns each input hypothesis a weight which determines its impact on the final combined translation.

Binary primary system model: The binary primary system model marks the words from the primary hypothesis (which determines the word order).

Language model: We train a 3-gram language model on the input hypotheses. This model extends the local word decisions and keeps the fluency from the input system.

Word penalty: The word penalty counts the number of words which basically means that this model is 1 for each arc.

Before adding the language model into the network, we need to enlarge the network so that each arc has its unique history. For a 3-gram language model, we need to generate a unique bigram history for each arc. This procedure enlarges the network. If we utilize a language model with a large history size, the lattice explodes and the decoding time increases dramatically. From our experience, taking a longer history than 2 into account does not improve the translation performance of the system combination.

The Jane system combination toolkit also provides the possibility to utilize additional models which are trained on external data which should be at hand as they were already needed for the input hypotheses generation. In this work, we integrated the optional usage of the following models:

Large language model: We use a large language model trained on larger monolingual target-side corpora. A large language model which is not only built on the input hypotheses should be at hand from the translation process of the individual systems.

IBM-1: Source-to-target and target-to-source IBM-1 lexical translation models obtained from bilingual training data can be used to keep a connection to the source sentence.

4.3.4 Decision Rule

Obtaining the best combined translation basically involves determining the shortest path[4] through the network. A network constructed by N input hypotheses (cf. Figure 4.7) consists of:

- Union of N confusion networks
- Each confusion network $n \in \{1,...,N\}$ has a length I_n
- Each position $i \in \{1,...,I_n\}$ in the confusion network has a choice of $K_{n,i}$ output words
- $e_{n,i,k}$ is the k-th arc in confusion network n at position i

Each arc $e_{n,i,k}$ is assigned one score $S(e_{n,i,k})$ which is a linear model combination of M different models $h_m(e_{n,i,k})$. Each of the model is weighted by its own scaling factor λ_m.

$$S(e_{n,i,k}) = \sum_{m=1}^{M} \lambda_m h_m(e_{n,i,k}) \tag{4.1}$$

We define the consensus translation for a single confusion network n as the sequence where at each position i the best word option $\hat{e}_{n,i}$ is selected from all possible word options $k \in \{1,..,K_{n,i}\}$ as given by the following equation:

$$\{e_{n,i,1}^{n,i,K_{n,i}}\} \rightarrow \hat{e}_{n,i}(\{e_{n,i,1}^{n,i,K_{n,i}}\}) := \underset{e_{n,i,k}:1\leq k\leq K_{n,i}}{\operatorname{argmin}} \{S(e_{n,i,k})\} \tag{4.2}$$

The combined best word sequence $\hat{e}_1^{\hat{I}}$ is extracted by the following decision rule:

$$\{\hat{e}_{n,i}\} \rightarrow \hat{e}_1^{\hat{I}}(\{\hat{e}_{n,i}\}) := \underset{\hat{e}_{n,1}^{n,I_n}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{I_n} S(\hat{e}_{n,i}) \right\} \tag{4.3}$$

The system combination implementation is part of the `Jane` toolkit which also includes a phrase-based and a hierarchical phrase-based decoder. We take use of the previously implemented optimization algorithms to optimize the λ_m weights. We utilize the MERT [Och 03] algorithm to optimize the free parameters in our experiments.

In Figure 4.2 a confusion network scored with some system weights is pictured. We used the shortest path algorithm to find the hypothesis with the highest score (the hypothesis along the path highlighted in dashed red). By using a linear model combination, we are able to add various new models into the network as we do in the following chapter.

4.4 Experimental Results

In this section, we present the experimental results of the latest official WMT system combination shared task.[5] We exclusively employ resources which were permitted for the constrained track of the task in all setups. The large language model is trained on News Commentary and Europarl data. The IBM-1 models are individually trained for each language pair on the data given by the organizers. As tuning set we use *newssyscombtune2011*, as test set we use *newssyscombtest2011*. Further, we utilize experiments on the recent BOLT Chinese→English as well as on the recent BOLT Arabic→English

[4] `Jane`'s implementation for building confusion networks is based on the OpenFST library [Allauzen & Riley$^+$ 07].

[5] The most recent system combination shared task that has been organized as part of the WMT evaluation campaign took place in 2011. `http://www.statmt.org/wmt11/system-combination-task.html`

corpora. Compared with the WMT experiments, the individual systems in BOLT are built on the same preprocessed data. As a result, the hypotheses of the individual systems are very similar and achieving substantial improvements with system combination is much more difficult. BLEU is the main metric in the WMT translation tasks and thus all systems are optimized against BLEU. The BOLT tasks are evaluated with HTER. HTER needs human interaction and consequently the team agreed to optimize the systems against BLEU-TER and only run HTER in the final evaluation. In addition to the comparison to other toolkits, we investigate the impact of the governed insertion (govIns) described in Section 4.3.2. All corpus statistics can be found in the appendix.

4.4.1 WMT German→English

Table 4.1 shows the results of the WMT German→English translation task. All system combination hypotheses are combinations of 6 input hypotheses. All single hypotheses (A-F) were generated during the WMT 2011 evaluation campaign. The `Jane` system combinations were generated in 2013 based on the same six input hypotheses of the WMT 2011 evaluation campaign. The best 2011 system combination submission outperforms the best single system by 2.1 points in BLEU and 2.1 points in TER. The baseline system combination setup including governed insertion (govIns) yields worse results compared to the best system combination submission of 2011. By applying an additional large language model, we achieve similar performance with 0.1 points better in BLEU and 0.1 points worse in TER. The application of IBM-1 translation models does not further improve the translation quality.

Table 4.1: German→English experimental results on the WMT 2011 system combination task. All single system hypotheses (A-F) were generated during the WMT 2011 evaluation campaign. The `Jane` system combination results were generated on the same single systems as the best 2011 system combination submission. The `Jane` system combination has been run in 2013.

system	**newssyscombtune2011**		**newssyscombtest2011**	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]
A	23.2	60.2	23.0	59.5
B	22.1	61.4	21.7	61.2
C	21.7	61.1	21.6	60.7
D	20.9	62.9	21.3	61.7
E	21.3	59.9	21.2	59.8
F	20.7	62.9	20.4	62.6
Best 2011 system combination submission	-	-	25.1	57.4
`Jane` system combination baseline	24.6	58.4	24.6	57.7
+ govIns	24.7	58.4	24.7	57.6
+ large LM	25.0	57.9	25.0	57.3
+ IBM-1	25.0	57.9	25.0	57.3

4.4.2 WMT Czech→English

The empirical evaluation of all the WMT Czech→English setups is presented in Table 4.2. For this task, we need to keep in mind that the best single system *System A* performs at least 8.3 BLEU points better than all other single systems. This is a scenario where we can expect system combination to underperform the best single system as the approach relies on majority voting. Thus, the best 2011 system combination setup yields only an improvement of 0.1 points in BLEU while losing 1.8 points in TER. The baseline setup yields the same performance in BLEU while only losing 0.2 points in TER. The advanced setups including the large language model and the IBM-1 translation models yield further improvements of 0.2 points in BLEU. Unfortunately, the TER increases by 0.9 points. Nevertheless, the system combination setups yield higher translation quality compared to the best 2011 system combination setup.

Table 4.2: Czech→English experimental results on the WMT system combination task. All single system hypotheses (A-D) were generated during the WMT 2011 evaluation campaign. The `Jane` system combination results were generated on the same single systems as the best 2011 system combination submission. The `Jane` system combination has been run in 2013. All system combinations combine 4 individual system outputs.

system	**newssyscombtune2011**		**newssyscombtest2011**	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]
A	27.4	55.1	28.7	53.4
B	22.5	61.3	22.4	60.8
C	21.2	62.7	20.3	62.7
D	18.7	64.1	19.7	63.0
Best 2011 evaluation system combination	-	-	28.8	55.2
`Jane` syscomb baseline	27.6	55.1	28.8	53.6
+ govIns	27.6	55.1	28.8	53.6
+ large LM	27.8	56.1	29.0	54.5
+ IBM-1	27.9	56.1	29.0	54.5

4.4.3 WMT French→English

The experimental results for the WMT French→English translation task are given in Table 4.3. In 2011, the best system combination submission achieves an improvement of 1.9 points in BLEU and 1.9 points in TER compared to the best single system. The baseline system combination setup yields the same performance in BLEU, but loses 0.2 points in TER compared to the best 2011 system combination submission. If we further extend the system combination and integrate a large language model as well as the IBM-1 models to the setup, we get additional gains of 0.2 points in BLEU and 0.2 points in TER. The advanced setup achieves better performance than the best 2011 evaluation submission. The governed insertion (govIns) has no impact on the translation performance for the WMT French→English translation task.

Table 4.3: French→English experimental results on the WMT system combination task. All single system hypotheses (A-H) were generated during the WMT 2011 evaluation campaign. The `Jane` system combination results were generated on the same single systems as the best 2011 system combination submission. The `Jane` system combination has been run in 2013. All system combinations combine 8 individual system outputs.

system	**newssyscombtune2011**		**newssyscombtest2011**	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]
A	29.8	52.5	29.4	52.0
B	29.7	51.8	29.1	51.4
C	29.7	53.0	28.8	52.9
D	29.3	52.9	28.5	52.8
E	28.4	53.3	28.3	53.0
F	28.8	27.4	28.0	53.3
G	27.4	54.6	27.2	53.8
H	27.2	53.5	26.7	52.9
Best 2011 evaluation system combination	-	-	31.3	50.1
`Jane` system combination baseline	32.2	50.7	31.3	50.3
+ govIns	32.2	50.7	31.3	50.3
+ large LM	32.4	50.5	31.4	50.0
+ IBM-1	32.4	50.3	31.5	50.0

4.4.4 WMT Spanish→English

Table 4.4 comprises all results of the empirical evaluation on the Spanish→English translation task. The best system combination submission of the WMT 2011 evaluation task improves translation quality by 3.5 points in BLEU and 1.3 points in TER compared to the best single engine. The system combination baseline further enhances translation quality by 0.2 points in BLEU while losing 0.6 points in TER. Applying all, the governed insertion algorithm, a large language model as well as the IBM-1 translation probabilities yield further improvements. In total the setup including all models outperforms the best system combination setup of 2011 by 0.7 points in BLEU and 0.1 points in TER.

4.4.5 BOLT Arabic→English

Table 4.5 shows the results of the BOLT Arabic→English translation task. In this task, all individual systems use the same preprocessing and are either phrase-based or hierarchical phrase-based machine translation engines which are partly enhanced with syntactical models. We used TER-BLEU as optimization criterion in the BOLT project. By combining all 5 individual systems and the governed insertion algorithm, we achieve improvements of 0.2 points in BLEU and 1.1 points in TER. A large language model as well as the IBM-1 translation probabilities do not further improve the translation quality. We present advanced system combination methods in the following chapters which are especially designed to yield higher gains for this task.

Table 4.4: Spanish→English experimental results on the WMT system combination task. All single system hypotheses (A-I) were generated during the WMT 2011 evaluation campaign. The Jane system combination results were generated on the same single systems as the best 2011 system combination submission. The Jane system combination has been run in 2013. All system combinations combine 9 individual system outputs.

system	newssyscombtune2011		newssyscombtest2011	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]
A	28.8	51.6	28.9	51.2
B	28.8	53.4	28.2	53.5
C	27.7	57.7	27.6	57.4
D	27.5	53.7	27.0	53.9
E	27.1	54.8	26.7	55.2
F	27.1	58.6	26.6	58.5
G	25.9	55.6	25.7	55.6
H	24.7	59.8	24.6	59.4
I	24.3	60.0	24.5	59.7
Best 2011 evaluation system combination	-	-	32.4	49.9
Jane syscomb baseline	33.6	50.2	32.6	50.5
+ govIns	33.6	50.0	32.7	50.3
+ large LM	33.6	49.9	32.9	50.3
+ IBM-1	33.8	49.5	33.1	50.0

Table 4.5: Results for the BOLT Arabic→English BOLT translation task. All system combinations are combinations of 5 individual systems.

system	tune		test	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]
A	25.4	57.1	27.4	56.9
B	25.7	56.6	27.1	57.6
C	25.8	56.5	27.0	56.9
D	26.1	55.8	26.4	57.1
E	25.2	57.3	26.2	58.0
Jane syscomb baseline	26.8	54.9	27.4	55.9
+ govIns	27.0	54.8	27.6	55.8
+ large LM	27.0	54.7	27.6	55.8
+ IBM-1	27.0	54.7	27.6	55.8

4.4.6 BOLT Chinese→English

The empirical evaluation of all the BOLT Chinese→English setups are presented in Table 4.6. We use TER-BLEU as optimization criterion in the BOLT project. We have similar conditions as in the WMT Czech→English translation task. System A has a much higher translation performance compared to all other individual systems. The system combination of all 7 individual systems yields a translation gain of 0.7 points in TER while losing 0.4 points in BLEU compared to the best single system. Further, the application of a large language model as well as the IBM-1 translation models only improves the translation quality by 0.1 points in BLEU. We present advanced methods in the following chapters which will help us to yield much higher gains for this kind of translation tasks where one single system outperforms the others.

Table 4.6: BOLT Chinese→English: All results are system combinations of 7 individual systems.

system	tune		test	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]
A	20.1	62.7	18.7	61.6
B	18.5	63.0	17.1	62.9
C	18.5	62.7	17.0	62.4
D	18.3	63.4	16.8	63.0
E	17.6	64.4	16.2	63.4
F	16.5	64.1	15.2	63.9
G	17.0	64.9	15.1	64.5
Jane syscomb baseline	20.1	61.4	18.3	60.9
+ govIns	20.1	61.4	18.3	60.9
+ large LM	20.1	61.3	18.4	60.9
IBM-1	20.1	61.3	18.4	60.9

4.5 Conclusion

RWTH's open source machine translation toolkit Jane now includes a state-of-the-art system combination framework. We found that the Jane system combination performs on a similar level or better than the best evaluation system combination submissions on all WMT 2011 system combination shared task language pairs (with English as target language). For all four language pairs we achieve improvements over the best 2011 evaluation system combination submission either in BLEU or TER. We achieve the highest improvement of 0.7 points in BLEU for Spanish→English when adding both the large language model and IBM-1 models. Adding the large language model over the baseline enhances the translation quality for all four language pairs. Adding IBM-1 lexicon models on top of the large language model is of marginal or no benefit for most language pairs, but at least provides slight improvements for Spanish→English. Applying the governed insertion algorithm improves the translation quality for Arabic→English, Spanish→English, and German→English while keeping translation quality for the other language pairs. In addition, we showed initial system combination results for the BOLT Arabic→English and BOLT Chinese→English translation tasks. The BOLT system combination tasks are more difficult as the individual systems share the same preprocessing and yield similar translations.

In the following chapters, we present advanced methods which address the given conditions and yield higher gains for both translations tasks, too.

5 Neural Network System Voting Model

In this chapter, we enhance the previous presented confusion network system combination approach with an additional model trained by a neural network. This chapter is motivated by the fact that the commonly used binary system voting models only assign each input system a global weight which is responsible for the global impact of each input system on all translations. This prevents individual systems with low system weights from having influence on the system combination output, although in some situations this can be helpful to improve the translation quality. Further, words which have only been seen by one or few systems rarely have a chance of being present in the combined output. In this chapter, we train a local system voting model by a neural network which is based on the words themselves and the combinatorial occurrences of the different system outputs. This gives system combination the option to prefer other systems at different word positions even for the same sentence. The work presented in this chapter has been published in [Freitag & Peter$^+$ 15].

5.1 Introduction

Adding more linguistic informed models (e.g. language model or translation model) additionally to the standard models into system combination seems to yield no or only small improvements. The reason is that all these models should have already been applied during the decoding process of the individual systems (which serve as input hypotheses for system combination) and hence already fired before system combination (cf. Table 4.5 or Table 4.6).

To gain further improvements with additional models, it is better to define models which are not used by an individual system. A simple model which can not be applied by any individual system is the binary system voting model (globalVote). This model is the most important one during system combination decoding as it determines the impact of each individual system. Each system i is assigned one globalVote model which fires if the word is generated by system i. Nevertheless, this simple model is independent of the actual word meaning and the score is only based on the global preferences of the individual systems. This disadvantage prevents system combination from producing words which have only been seen by systems with low system weights (low globalVote model weights). To give systems and words with low weights a chance to affect the final output, we define a new local system voting model (localVote) which makes decisions based on the current word options and not only on a general weight. The local system voting model allows system combination to prefer different system outputs

at different word positions even for the same sentence.

Motivated by the success of neural networks in language modelling [Bengio & Schwenk$^+$ 06, Schwenk & Gauvain 02] and translation modelling [Son & Allauzen$^+$ 12], we choose feedforward neural networks to train the novel model. Instead of calculating the probabilities in a discrete space, the neural network projects the words into a continuous space. This projection gives us the option to assign probability also to input sequences which were not observed in the training data. In system combination each training sentence has to be translated by all individual system engines which is time consuming. Due to this we have a small amount of training data and thus it is likely that many input sequences of a test set have not be seen during training.

The remainder of this chapter is structured as follows: in Section 5.2, we introduce additional related work. In Section 5.3, the novel local system voting model is introduced. In Section 5.4, experimental results are presented which are analyzed in Section 5.5. In Section 5.6, we present three translation examples which demonstrate the advantages of the novel model. Finally, we conclude this chapter in Section 5.7.

5.2 Related Work

All system combination approaches presented in section 4.2 only use the global system voting models. Regarding to this chapter, there has been similar effort in the area of speech recognition. We introduce one additional publication to this chapter:

[Hillard & Hoffmeister$^+$ 07] Similar work has been presented for system combination of speech recognitions systems: the authors train a classifier to learn which system should be selected for each output word. The learning target for each slot is the set of systems which match the reference word, or the null class if no systems match the reference word. Their novel approach outperforms the ROVER baseline by up to 14.5% relatively on an evaluation set.

5.3 Novel Local System Voting Model

In the following subsections we introduce a novel local system voting model (localVote) trained by a neural network. The purpose of this model is to prefer one particular word sequence in the confusion network and therefore all local word decisions between two nodes leading to this particular word sequence. More precisely, we want the neural network to learn an oracle word sequence extracted from the confusion network graph which leads to the lowest error score. In Subsection 5.3.1, we describe a polynomial approximation algorithm to extract the best sentence level BLEU (sBLEU) word sequence in a confusion network. Taking this word sequence as reference word sequence, we define the model in Subsection 5.3.2 followed by its integration in the linear model combination in Subsection 5.3.3.

5.3.1 Finding sBLEU-Optimal Hypotheses

In this section, we describe a polynomial approximation algorithm to extract the best sBLEU hypothesis from a confusion network. [Leusch & Matusov$^+$ 08] showed that this problem is generally NP-hard for the popular BLEU [Papineni & Roukos$^+$ 02] metric. Nevertheless, we need some word sequence which serves as "reference word sequence".

Using BLEU as metric to extract the best possible word sequence is problematic as in the original BLEU definition there is no smoothing for the geometric mean. This has the disadvantage that the BLEU score becomes zero already if the four-gram precision is zero, which can happen obviously often with short or difficult translations. To allow for sentence-wise evaluation, we use the sBLEU

metric [Lin & Och 04], which is basically BLEU where all n-gram counts are initialized with 1 instead of 0. The brevity penalty is calculated only on the current hypothesis and reference sentence.

We use the advantage that confusion networks can be sorted topologically. We walk the confusion network from the start node to the end node, keeping track of all n-grams seen so far. At each node we keep a k-best list containing the partial hypotheses with the most n-gram matches leading to this node and recombine only partial hypotheses containing the same translation. As the search space can become exponentially large, we only keep k possible options at each node. This pruning can lead to search errors and hence yield non-optimal results. If needed for hypotheses with the same n-gram counts, we prefer hypotheses with a higher translation score based on the original models. For the final node we add the brevity penalty to all possible translations.

As we are only interested in arc decisions which match a reference word, we simplify the confusion network before applying the algorithm. If all arcs between two adjacent nodes are not present in the reference, we remove all of them and add a single arc labeled with "UNK". This reduces the vocabulary size and still gives us the same best SBLEU scores as before. In Figure 5.1, a confusion network of four input hypotheses is given. As the words *black*, *red*, *orange*, and *green* are all not present in the reference, all of them are mapped to one single "UNK" arc (cf. Figure 5.2). The best SBLEU word sequence is *the UNK car*.

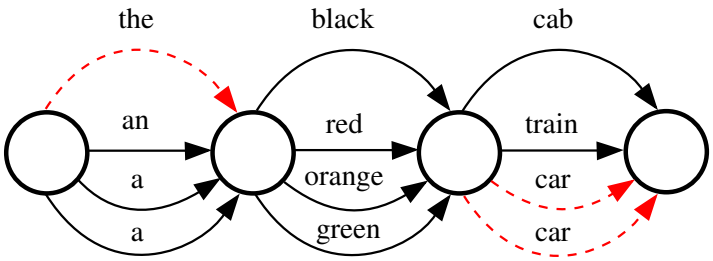

Figure 5.1: System A: *the black cab* ; System B: *an red train* ; System C: *a orange car* ; System D: *a green car* ; Reference: *the blue car* .

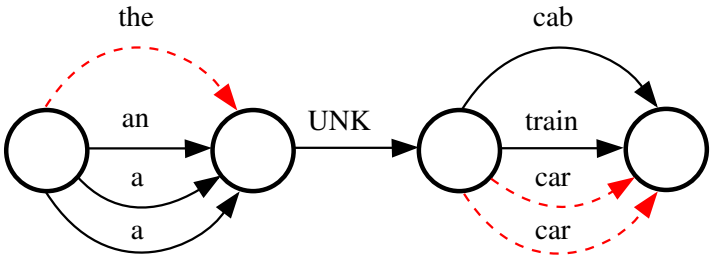

Figure 5.2: As the words *black*, *red*, *orange*, and *green* in Figure 5.1 are all not present in the reference (*the blue car*), they are mapped to one single "UNK" arc.

5.3.2 Model Training

The purpose of the new localVote model is to prefer the best SBLEU word sequence and therefore to learn the word decisions between all adjacent nodes which lead to this particular word sequence. During the extraction of the best SBLEU hypotheses from the confusion network, we keep track of all arc decisions. This gives us the possibility to generate local training examples based only on the I arcs between two nodes. For the confusion network illustrated in Figure 5.2, we generate two training

examples for the neural network training. Based on the arcs *the*, *an*, *a*, and *a* we learn the output *the*. Based on the arcs *cab*, *train*, *car*, and *car* we learn the output *car*.

In all upcoming system setups, we utilize the open source toolkit NPLM [Vaswani & Zhao$^+$ 13] for training and testing the neural network models. We use the standard setup as described in the paper and use the neural network with one projection layer and one hidden layer. For more details we refer the reader to the original paper of the NPLM toolkit. The inputs to the neural network are the I words produced by the I different individual systems. The outputs are the posterior probabilities of all words of the vocabulary. The input uses 1-of-n coding, i.e. the i-th word of the vocabulary is coded by setting the i-th element of the vector to 1 and all the other elements to 0.

For a system combination of I individual systems, a training example consists of $I+1$ words. The first I words (input of the neural network) are representing the words of the individual systems, the last position (output of the neural network) serves as slot for the decision we want to learn (extracted from the best SBLEU word sequence). We do not add the "UNK" arcs to the neural network training as they do not help to increase the SBLEU score. Figure 5.3 shows the neural network training example for the last words of Figure 5.2. The output of each individual system provides one input word. In Table 5.1 the two training examples for Figure 5.2 are illustrated.

As a neural network training example only consists of the I words between two adjacent nodes, we are able to produce much more training examples than having sentences. For a system combination of I systems and a development set of S sentences with an average sentence length of L, we can generate $I*S*L$ neural network training examples.

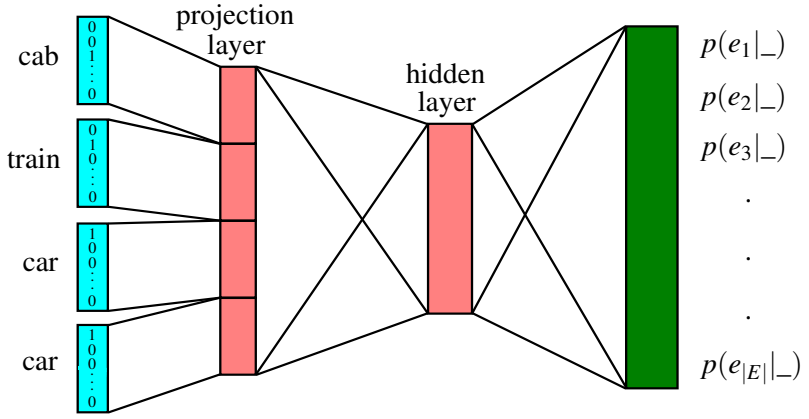

Figure 5.3: Unigram neural network training example: System A produces *cab*, System B *train*, System C *car*, System D *car*, reference is *car*. We apply 1-of-n encoding to map words to a suitable neural network input.

Table 5.1: Training examples from Figure 5.2. The output of each individual system provides one input word, be it a word or ε. The reference is the word selected by the best SBLEU word sequence.

input layer				output layer
system A	system B	system C	system D	reference
the	an	a	a	the
cab	train	car	car	car

Table 5.2: Training examples (bigram) from Figure 5.2. In addition to the current words, the predecessor words are taken into account.

input layer				output layer
system A	system B	system C	system D	reference
<s>the	<s>an	<s>a	<s>a	the
black cab	red train	orange car	green car	car

Further, we can expand the model to use arbitrary history size, if we take the predecessor words into account. Instead of just using the local word decision of a system, we add additionally the predecessors of the individual systems into the training data. In the example, we utilize the bigram *red train* instead of the unigram *train* for system *B* into the training data. In Figure 5.4 one example is illustrated. In Table 5.2 all bigram training examples of Figure 5.2 can be seen.

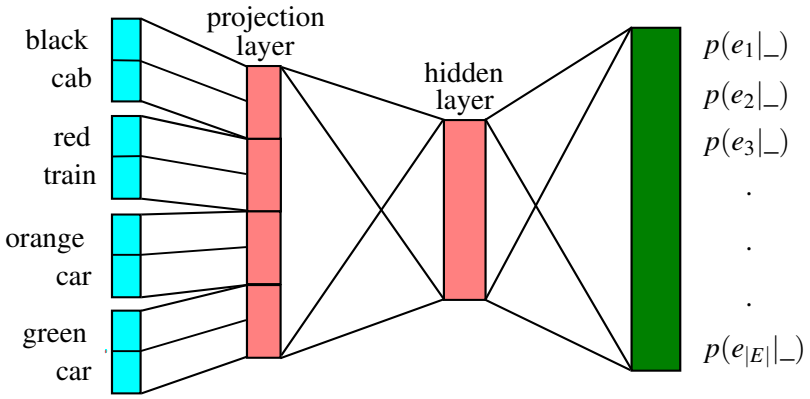

Figure 5.4: Bigram neural network training example: System A produces *black cab*, System B *red train*, System C *orange car*, System D *green car*, reference is *car*. We apply 1-of-*n* encoding to map words to a suitable neural network input format.

5.3.3 Model Integration

Having a trained localVote model, we then add it as an additional model $h_m(e_{n,i,k})$ to the linear model combination of the previous Chapter 4. For a network constructed by N input hypotheses, we assign each arc $e_{n,i,k}$ at position i of confusion network n and option k one additional score $h_m(e_{n,i,k})$:

$$S(e_{n,i,k}) = \sum_{m=1}^{M} \lambda_m h_m(e_{n,i,k}) \tag{5.1}$$

We calculate for each arc the probability of the word in the trained neural network. E.g. for Figure 5.1, we extract the probabilities for all arcs by the strings illustrated in Table 5.3. Finally, we add the scores as a new model to the linear framework and assign it a weight which is trained additionally to the standard model weights with MERT.

Table 5.3: Calculating the probability for all possible output words from Figure 5.1. The output layer is the current generated word.

input layer				output layer
system A	system B	system C	system D	arc word
the	an	a	a	the
the	an	a	a	an
the	an	a	a	a
black	red	orange	green	black
black	red	orange	green	red
black	red	orange	green	orange
black	red	orange	green	green
cab	train	car	car	cab
cab	train	car	car	train
cab	train	car	car	car

5.3.4 Word Classes

The neural network training sets are small as all sentences have to be translated by all individual system engines. This results in many unseen words in the test sets. To overcome this problem, we use word classes [Och 99] instead of words which were trained on the target part of the bilingual training corpus in some experiments. We utilize the trained word classes on both input layer and output layer.

5.4 Experiments

All experiments have been conducted with the novel system combination approach presented in Chapter 4. For training and scoring neural networks, we use the open source toolkit NPLM [Vaswani & Zhao$^+$ 13]. NPLM is a toolkit for training and using feedforward neural language models. Variations in neural network architecture have been tested. We tried different hidden layer sizes as well as projection layer sizes. We achieved similar results for all setups and decided to stick to 1 hidden layer whose size is 200, a learning rate of 0.08 and let the training run 20 epochs in all experiments.

Translation quality is measured in lowercase with BLEU [Papineni & Roukos$^+$ 02] and TER [Snover & Dorr$^+$ 06] whereas the performance of each setup is the best score on the tune set across five different MERT runs. The system combination weights (Equation 4.1) of the linear model are optimized with MERT on 200-best lists with (TER-BLEU)/2 as optimization criterion. For all language pairs we use three different test sets. In the following the test set for extracting the training examples for the neural network training is labeled as *tune (NN)*. The test set *tune (MERT)* indicates the tune set for MERT and *test* indicates the blind test set.

The individual systems are different extensions of phrase-based or hierarchical phrase-based systems. The systems are built on the same amount of preprocessed training data and differ mostly in the models which are used to score the translation options. Further, some systems are syntactical augmented based on syntax trees on either source or target side.

5.4.1 BOLT Chinese→English

For Chinese→English, we use the current BOLT data set (corpus statistics are given in Table 10.5). The test sets consist of text drawn from "discussion forums" in Mandarin Chinese. We utilize nine individual systems to perform the system combination experiments. The lambda weights are optimized on a tune set of 985 sentences (tune (MERT)). We train the proposed localVote model on 15,323,897 training examples extracted from the 1844 sentences tune (NN) set.

As a first step we have to determine the k-best pruning threshold for extracting the sBLEU optimal word sequence from the current confusion networks (cf. Section 5.3.1). In Figure 5.5 the (TER-BLEU)/2 results of the sBLEU optimal hypotheses extracted with different k-best sizes are given. Although, the BLEU score improves by setting k to a higher value, the computational time increases. To find a tradeoff between running time and performance, we set the k-best size to 1200 in the following experiments.

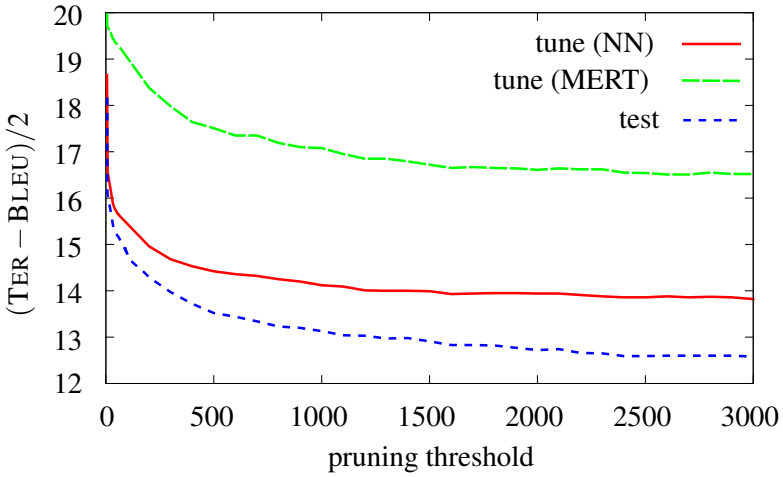

Figure 5.5: BOLT Chinese→English: $(\text{TER} - \text{BLEU})/2$ scores for different k-best pruning thresholds. We set k to 1200 in all our following experiments.

Experimental results are given in Table 5.4. The *baseline* is a system combination run without any localVote model of nine individual systems using the standard models as described in section 4.3.3. The *oracle* score is calculated on the hypothesis of the sBLEU best word sequence extracted with $k = 1200$. We train the neural network on 15,323,897 training examples generated from the 1844 tune (NN) sentences. By training a neural network model based on unigram decisions, we gain small improvements of 0.6 points in TER. As we have only a small amount of training data, many words have not been seen during neural network training. To overcome this problem, we train 1500 word classes on the target part of the bilingual data. Learning the localVote model on word classes gain improvements of 0.7 points in BLEU and 0.6 points in TER. By taking a bigram history into the training of the neural network, we reach only small further improvement. Compared to the *baseline*, the *bigram neural network model* outperforms the *baseline* by 0.3 points in BLEU and 0.6 points in TER. By using word classes, we gain improvement of 0.4 points in BLEU and 1.0 points in TER.

All results are reached with a word class size of 1500. In Figure 5.6 the $(\text{TER} - \text{BLEU})/2$ scores of system combinations including one unigram localVote model trained with different word class sizes are illustrated. Independent of the word class size, system combination including a localVote model always performs better compared to the baseline. The best performance is reached by a word class size of 1500. One reason for the loss of performance when using no word classes is the size of the neural network tune set. Within a size of 1844 sentences, many words of the test set have never been

Table 5.4: Results for the BOLT Chinese→English translation task. The *baseline* is generated with the standard set of models as described in Chapter 4. Each model is trained once with and once without word classes on both input and output layer of the neural network.

system combination	word classes in model training	tune		test	
		BLEU[%]	TER[%]	BLEU[%]	TER[%]
baseline (Chapter 4)		17.9	61.5	18.3	60.9
+unigram neural network model	**no**	18.1	61.2	18.3	60.3
	yes	18.4	61.5	**19.0**	60.3
+bigram neural network model	**no**	18.1	61.3	18.6	60.3
	yes	18.1	61.2	**18.7**	**59.9**
oracle word sequence		28.6	62.3	31.1	57.2

seen during neural network training. The test set has a vocabulary size of 6106 within 2487 words (40.73%) are not present in the training set (tune (NN)) of the neural network. For the MERT tune set 2556 words (40.91%) are not present in the neural network training set. Word classes tackle this problem and it is much more likely that each word class has been seen during the training procedure of the neural network.

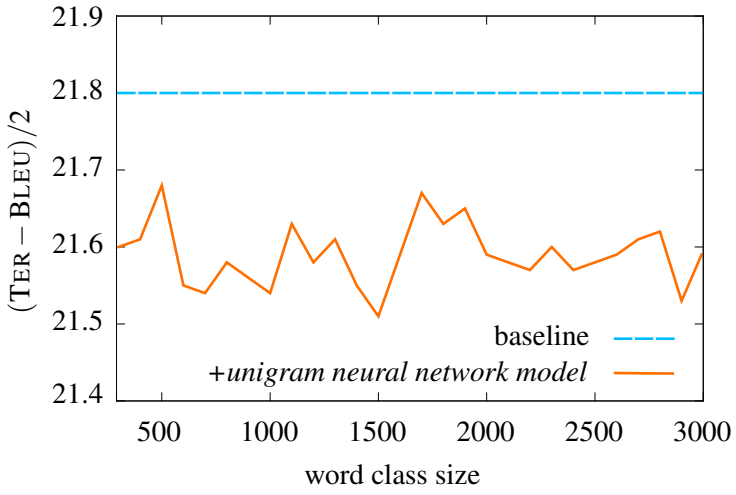

Figure 5.6: BOLT Chinese→English: (TER − BLEU)/2 tune set scores for different word class sizes. We set the word class size to 1500 in all our experiments.

5.4.2 BOLT Arabic→English

For Arabic→English, we use the current BOLT data set (corpus statistics are given in Table 10.6). The test sets consist of text drawn from ”discussion forums” in Egyptian Arabic. We train the neural network on 6,591,158 training examples extracted from the 1510 sentences tune (NN) dev set. The model weights are optimized on a 1080 sentences tune set. All results are system combinations of five individual systems. The test set has a vocabulary size of 3491 within 1510 words (43.25%) are not present in the training set (tune (NN)) of the neural network. For the MERT tune set 1549 words

(43.24%) are not part of the neural network training set.

We run the same experiment pipeline as for Chinese→English and first determine the k-best threshold for getting the oracle word sequence in the confusion networks. As the Arabic→English system combination is only based on 5 individual systems, the confusion networks are much smaller. We set the pruning threshold to 1000 ($k = 1000$) which is a good tradeoff between running time and performance. Figure 5.7 shows the $(\text{TER} - \text{BLEU})/2$ scores for different k-best pruning thresholds. Increasing k to a higher value then 1000 improves the $(\text{TER} - \text{BLEU})/2$ only slightly.

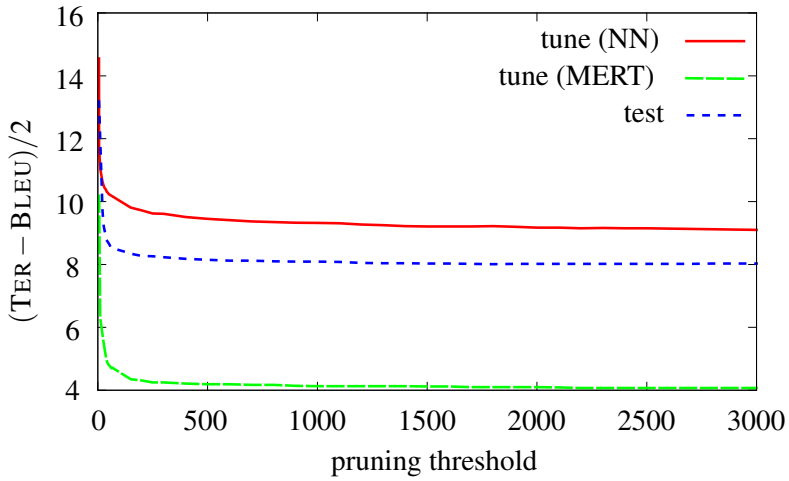

Figure 5.7: BOLT Arabic→English: $(\text{TER} - \text{BLEU})/2$ scores for different k-best pruning thresholds. We set the pruning threshold to 1000 ($k = 1000$) which is a good tradeoff between running time and performance.

Experimental results are given in Table 5.5. The *baseline* is a system combination run without any localVote model of five individual systems using the standard models as described in section 4.3.3. The *oracle* score represents the score of the sBLEU best word sequence extracted with $k = 1000$. Training a localVote *unigram neural network model* based on the best sBLEU word sequence gives us improvement of 0.9 points in BLEU compared to the *baseline*. Adding bigram context to the neural network training yields improvement of 0.8 points in BLEU compared to the *baseline* system combination. By training word classes on the bilingual part of the training data, we gain additional improvements. By using word classes in both input layer and output layer of the *bigram neural network model*, we reached the best performance with 1.1 points in BLEU compared to the *baseline* setup.

All results are conducted with a word class size of 1000. The tune set performance of different unigram localVote models trained on different word class sizes are illustrated in Figure 5.8. The results are fluctuating and we set the word class size to 1000 in all Arabic→English experiments.

5.5 Analysis

In this section we compare the final translations of the Chinese→English system combination *+bigram wcNN* with the *baseline*. The word occurrence distributions for both setups are illustrated in Table 5.6. This table shows how many input systems produce a certain word and finally if it is part of the system combination output. As the original idea of system combination is based on majority voting, it should be more likely that a word which is produced by more input systems is in the final system combination output than a word which is only produced by few input systems. E.g. 11008 words have been produced by all 9 individual systems from which all of them are in both the system

Table 5.5: Results for the BOLT Arabic→English translation task. The *baseline* is generated with the standard set of models as described in Chapter 4. Each model is trained once with and once without word classes on both input and output layer of the neural network.

system combination	word classes in model training	tune		test	
		BLEU[%]	TER[%]	BLEU[%]	TER[%]
baseline (Chapter 4)		30.1	51.2	27.6	55.8
+unigram neural network model	**no**	31.4	51.2	28.5	56.0
	yes	31.1	51.1	28.3	55.7
+bigram neural network model	**no**	31.3	51.1	28.4	55.8
	yes	31.4	51.2	**28.7**	56.0
oracle word sequence		38.1	46.3	34.8	50.9

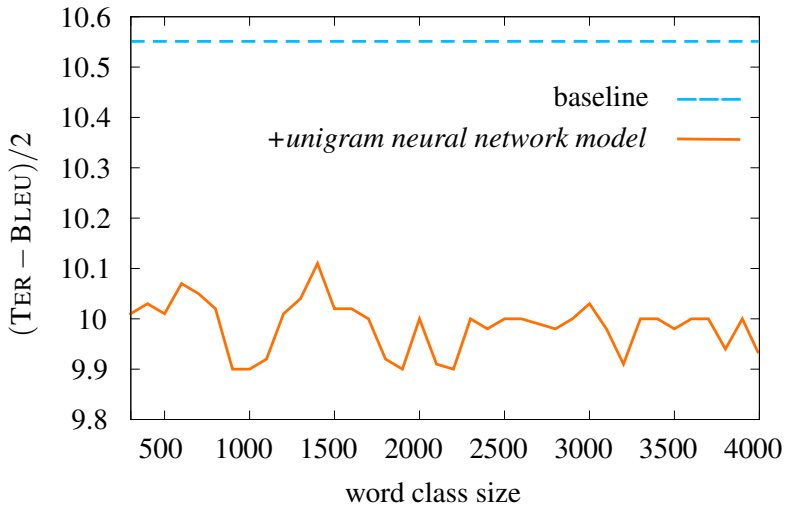

Figure 5.8: BOLT Arabic→English: (TER − BLEU)/2 tune set scores for different word class sizes. We set the word class size to 1000 in all Arabic→English experiments.

combination *baseline* and the advanced system *+bigram wcNN*. If a word is only produced by 8 individual systems, a ninth system does not produce this word. 98,9% of the words produced by only 8 different individual systems are in the final *baseline* system combination output. The missing words result mostly from alignment errors produced by the pairwise alignment algorithm when aligning the single systems together.

We observe the problem that the globalVote models prevent words that have only been produced by few systems to be present in the system combination output. In Table 5.6, you can see that words which are only produced by 1-4 individual systems are more likely to be present in the final output when including the novel localVote model. For example, 592 of 6129 words that have only been produced by two individual systems are in the output of the *baseline* setup. Whereas the advanced *+bigram wcNN* setup contains additional 172 words. These statistics demonstrate the functionality of the novel localVote model, which does not only improve the translation quality in terms of BLEU, but also tackles the problem of the dominating globalVote models.

Table 5.6: Word occurrence distribution for the Chinese→English setup. First column indicates in how many systems a word appears. E.g. 120/14072 (0.9%) indicates that 14072 words only appear in one individual input system from which 120 (0.9%) are present in the baseline system combination hypothesis.

#	*baseline*		*+bigram neural network model with word classes*	
1	120/14072	(0.9%)	214/14072	(1.5%)
2	592/ 6129	(9.7%)	764/ 6129	(12.5%)
3	1141/ 4159	(27.4%)	1319/ 4159	(31.7%)
4	1573/ 3241	(48.5%)	1669/ 3241	(51.5%)
5	2051/ 2881	(71.2%)	1993/ 2881	(69.2%)
6	2381/ 2744	(86.8%)	2332/ 2744	(85.0%)
7	2817/ 2965	(95.0%)	2820/ 2965	(95.1%)
8	3818/ 3860	(98.9%)	3815/ 3860	(98.8%)
9	11008/11008	(100.0%)	11008/11008	(100.0%)
$\sum$	25537/51059	(50.0%)	25977/51059	(50.8%)

The Arabic→English word occurrence distribution is illustrated in Table 5.7. A similar scenario as for the Chinese→English translation task can be observed. The words which only occur in few individual systems have a much higher chance to be in the final output when using the novel local voting system model. It is also visible that the neural network model prevents some words of being in the combined output even if the word have been produced by 4 of 5 systems. The novel local system voting model gives system combination the option to select words which have only been generated by few individual systems.

Table 5.7: Word occurrence distribution for the Arabic→English setup. First column indicates in how many systems a word appears. E.g. 214/5791 (3.7%) indicates that 5791 words only appear in one individual input system from which 214 (3.7%) are present in the baseline system combination hypothesis.

#	*baseline*		*+bigram neural network model with word classes*	
1	214/ 5791	(3.7%)	285/ 5791	(4.9%)
2	1225/ 3200	(38.3%)	1243/ 3200	(38.8%)
3	2162/ 2719	(79.5%)	2297/ 2719	(84.5%)
4	3148/ 3207	(98.2%)	3119/ 3207	(97.3%)
5	14602/14602	(100.0%)	14602/14602	(100.0%)
$\sum$	21351/29526	(72.3%)	21546/29526	(73.0%)

5.6 Translation Examples

In this section, we present three translation examples which show the improvement of the novel local system voting model. All examples are extracted from the BOLT Chinese→English setup. We compare the *baseline* and the *+bigram neural network model* setup including word classes. All scores are measured with TER and sBLEU. To allow for sentence-wise evaluation, [Lin & Och 04] define the sBLEU metric which is basically BLEU where all *n*-gram counts are initialized with 1 instead of 0. The brevity penalty is calculated only on the current hypothesis and reference sentence.

In Table 5.8, the first translation example is given. The period is only produced by one single system. The new localVote model gives confusion network decoding the option to use the full stop in the final output, even if it only has been generated by one system. In the second example (Table 5.8), the word "recording" is only produced by one individual system. The system combination including the new localVote model produced a much better translation in terms of TER and sBLEU.

Table 5.8: Translation examples extracted from the BOLT Chinese→English translation task. We compare the baseline confusion network approach with the advanced approach that includes *+bigram neural network model* and word classes. The translation scores are: TER: 75.00 sBLEU: 37.79 (target-based), TER: 0.00 sBLEU: 100.00 (source-based).

reference	sudan is divided .
baseline	sudan has been divided
+bigram NN model	sudan is divided .

Table 5.9: Translation examples extracted from the BOLT Chinese→English translation task. We compare the baseline confusion network approach with the advanced approach that includes *+bigram neural network model* and word classes. The translation scores are: TER: 68.42 sBLEU: 14.52 (target-based), TER: 47.37 sBLEU: 36.60 (source-based).

reference	i watched the recording on al arabiya . does he think we are crazy , naive or what ?
baseline	i saw the car did n't we are idiots or fools or what .
+bigram NN model	i saw the recording on the car did n't think we are idiots or fools or what ?

5.7 Conclusion

In this work we proposed a novel local system voting model (localVote) which has been trained by a feedforward neural network. In contrast to the traditional globalVote model, the presented localVote model takes the word contents and their combinatorial occurrences into account and does not only promote global preferences for some individual systems. This advantage gives confusion network decoding the option to prefer other systems at different positions even in the same sentence. As all words are projected to a continuous space, the neural network gives also unseen word sequences a useful probability. Due to the relatively small neural network training set, we used word classes in some experiments to tackle the data sparsity problem.

Experiments have been conducted with high quality input systems for the BOLT Chinese→English and Arabic→English translation tasks. Training an additional model by a neural network with word classes yields translation improvement of 0.9 points in BLEU and 0.5 points in TER. We also took

word context into account and added the predecessors of the individual systems to the neural network training which yield additional small improvement compared to an unigram based localVote model. We analyzed the translation results and the functionality of the localVote model. The occurrence distribution shows that words which have been produced by only few input systems are more likely to be part of the system combination output when using the proposed localVote model.

6 Source-Aligned MT System Combination

Despite translation quality improves, confusion network system combination is independent of the source sentence and is only based on the target words. In this chapter, we investigate the idea of using the source alignment information of each individual input hypothesis to align the system outputs. We substitute the general word-to-word alignment by a phrase-to-phrase alignment, which enhances confusion network decoding with the option to model phrase alternatives of different length. By that, we fix several alignment errors which occur by ignoring the source sentence. Experiments on the BOLT Chinese→English and Arabic→English data yield improvements of 0.6 points in BLEU and 0.5 points in TER. In addition, human analysis of the resulting lattices reveals that the lattices produced with source information are superior to the traditional confusion networks.

6.1 Introduction

Research in machine translation system combination mostly focuses on switching the alignment techniques for calculating the pairwise word alignment between the input systems. Although, several successful extensions have been developed, all approaches share one major weakness: they are all based only on the individual translations without looking at the source sentence and the phrase alignment from the input systems. One reason is the requirement to provide additional to the translations the corresponding phrase information used in the decoding process of each individual system. As part of this thesis, we explore alignment techniques that are based on the phrase alignments of the individual input systems. Further, instead of generating a word-to-word confusion network, we allow for phrase alternatives of different length.

There are several shortcomings by using a word-to-word alignment without any source information in confusion network system combination. First of all, it is impossible to generate m-to-n alignments and therefore impossible to align phrases with different lengths. Further, as the words have no connection to the source sentence, there is no guarantee that alternative translations of the same source words are aligned. It may happen that the combined translation contains more than one translation of the same source sequences. Even worse, we can lose some information and skip the translations of a source word. In our novel algorithm, a strong connection to the source sentence is kept and as a consequence we are able to align very different translation outputs and generate a more robust lattice. In this chapter, we present improvements in translation quality and compare the novel lattices

with the conventional confusion networks generated by either the GIZA++ or the METEOR alignment algorithm.

This chapter is organized as follows. We start with a short comparison to the most similar work in Section 6.2. In Section 6.3, we present the novel lattice generation which is based on the phrase alignment of the individual input systems. In Section 6.4, the shortcomings by applying confusion network decoding without any source information are illustrated. The system outputs of confusion network decoding and lattice decoding with source information are evaluated both automatically (Section 6.5) and manually (Section 6.6). Translation examples are illustrated and discussed in Section 6.7. We conclude this chapter in Section 6.8.

6.2 Related Work

The confusion network system combination approaches presented in section 4.2 only align the hypotheses based on the target words. There are some publications which either tackle the problem of missing *m*-to-*n* alignments or extend system combination by using the internal phrase information of the decoders. Different to the related work, we use both benefits and introduce in addition a novel lattice generation algorithm.

[Rosti & Ayan$^+$ 07] use the internal phrase alignment information to build a new phrase table and redecode the source sentence with the new phrase table. The most common fact to this chapter is the usage of the phrase alignment, even though the usage is different. The authors come to the conclusion that the general confusion network approach outperforms their re-decoding approach. This result from the fact that re-decoding allows for flexible word reorderings. Our proposed approach only allows reorderings seen by the individual systems themselves.

[Feng & Liu$^+$ 09] build a phrase-to-phrase lattice from the individual system outputs. Similar to confusion network system combination, the authors ignore the internal phrase alignment and build their lattice on paraphrases learned only on the individual translations. Further, they stick to one backbone translation which defines the word order of the final translations. We allow for combinations of word orders from the individual systems while keeping a strong connection to the source sentence.

[Du & Ma$^+$ 09] use the internal phrase alignment information of the individual system decoders and employ on top of that a mapping strategy and normalization model to acquire only 1-to-1 alignment links. The authors build the confusion networks in the same manner as in the original confusion network system combination approach based on the previous generated 1-to-1 alignments. Similar to this chapter, the authors use the phrase alignment information from the individual systems. Instead of reducing the alignments to 1-to-1 alignments, we also use *n*-to-*m* alignments to capture translations of different lengths. Contrary to the authors, we built up a new lattice which overcomes the restrictions of confusion networks.

6.3 Source-Aligned System Combination

In this section, we present a novel source-aligned lattice system combination approach which is based on the phrase decoding information from the individual system engines. In the following sections, this approach is referred as *source-aligned system combination*. The confusion network system combination approach using GIZA++ [Och & Ney 03] or METEOR [Denkowski & Lavie 14] for the alignment procedure is referred as *target-based system combination*.

6.3.1 Source-Aligned Lattice

The construction of a source-aligned lattice is described with the help of two translation examples of the German source sentence *Jan gab seinem Vater Bücher*. Both translations are illustrated in Figure 6.1. Translation 1 *his father gave Jan books* consists of four different phrase alignments, translation 2 *Jan gave dad journals* has a monotone phrase alignment. Different to the confusion network alignments, we also have *n*-to-*m* alignments.

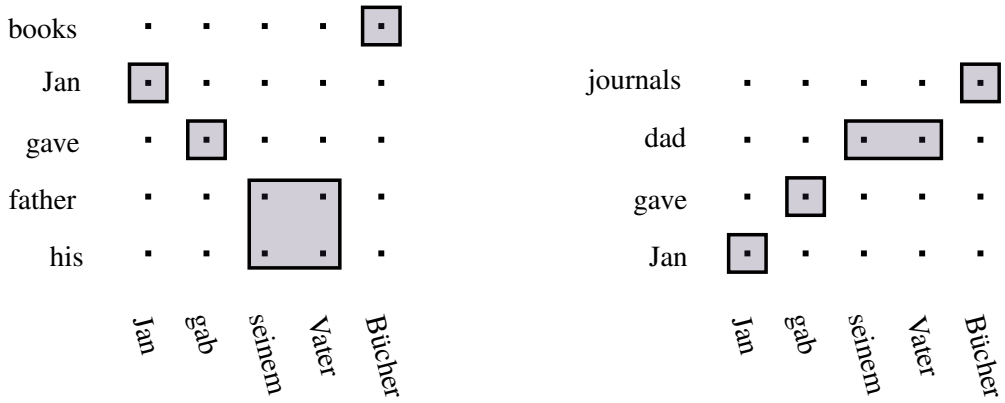

Figure 6.1: Two different translations of the source sentence *Jan gab seinem Vater Bücher*. The source phrase "seinem Vater" is an *n*-to-*m* alignment in both translations.

In Table 6.1 all phrases of the two input hypotheses are listed. The source indices are defined by the indices of the source words of the phrase. These eight phrases are the only needed information to build the final lattice.

Table 6.1: All phrases given by the two different input hypotheses from Figure 6.1.

system	source indices	translation
1	2,3	his father
1	1	gave
1	0	Jan
1	4	books
2	0	Jan
2	1	gave
2	2,3	dad
2	4	journals

We build the source-aligned lattice with the following algorithm. The lattice is initialized with one start node, labeled with the empty set which means that at this node no source word has been translated yet. As next step all individual system translations (input hypotheses) are sequentially inserted one after the other. We insert the phrases from each input hypothesis in the same order as it appears in its translation. For the first phrase, we insert an arc leading from the start node to a new node labeled with the source indices translated by the actual phrase. In Figure 6.2, the first phrase of translation 1 has been inserted. The arc is labeled with its translation "his father" and yields from the start node to a new node labeled with the indices of its source words.

For the next phrases, we insert arcs leading from the node labeled with all previous translated source positions to a new node labeled with the new total translated source positions. The final node of the last phrase of each sentence is marked as a final node (as all source words are already translated). In Figure 6.3, all phrases of system 1 has been inserted into the lattice in the same order as in its original translation.

The same procedure is applied for all N input hypotheses. The lattice contains N non overlapping paths and still consists of only N translation options. To give lattice decoding the power to generate new hypotheses, we merge all nodes having the same label and thus the same source words translated. In Figure 6.4, the lattice containing both translations is illustrated. Not only the original translations can be extracted, but also four new combined translation options have been emerged. The start node is the empty set, as no source word has been translated. On the final node all source words are translated.

We can already run a decoding process on the current lattice and even generate new combined translations. However, we first want to insert more meaningful translation options and generate a valuable network. The translation between two nodes cover a set of source words. If there exist phrases which cover the same source positions, we can insert them as alternative phrase translation options. In Figure 6.5 the lattice including new translation options is illustrated. E.g. the source positions 2 and 3 can be translated by either ”his father” or ”dad”. As both phrases cover the same source words, we add both as alternative phrases into the lattice (cf. Table 6.1).

It is possible that two disjunct paths leading from the same node A to the same node B contain the same translation, because more than one system possibly produce the same translation for the same source span. We are only interested in the final translation and merge all arcs that connect the same nodes and are labeled with the same translation. We first split the phrases into words (for an arc labeled with two words, a new node is needed to be inserted) and then make the lattice deterministic. Modifying the lattice to be deterministic only merges identical arcs, but does not generate new translation options. Figure 6.6 illustrates the final lattice.

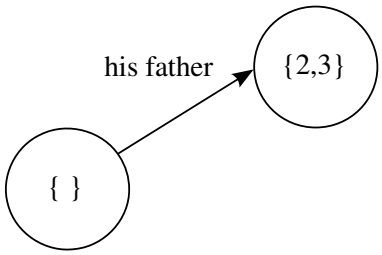

Figure 6.2: Lattice generation: each node in the source-aligned lattice corresponds to a set of source words that have been translated (coverage vector). An arc corresponds to a translation of one or more source words, be it a word or a longer phrase. All paths leading to the same node have exactly translated the same set of source words.

6.3.2 Non-Aligned Words

Some source words have no translation in the target language and are not aligned to any target word. These non-aligned words need a special handling as otherwise the full translation does not cover all source positions. Before generating the source based lattice, we check all sentences for non-aligned words. For each non-aligned source word w_i, we add an additional alignment point to the phrase containing word w_{i-1}. This assures that all translations end in the same final node.

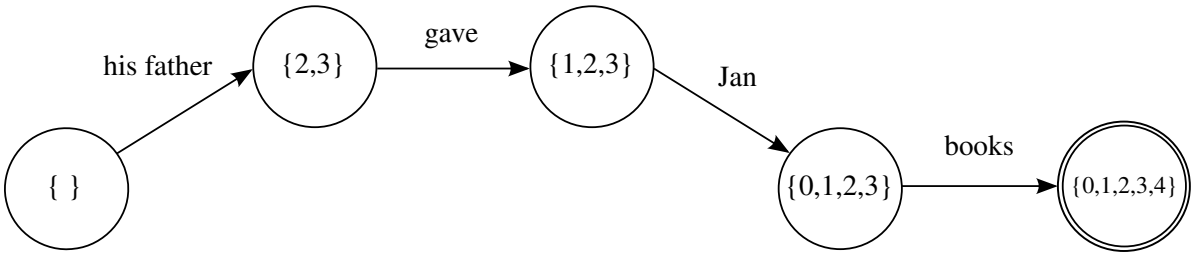

Figure 6.3: Lattice generation: translation 1 has been inserted into the network based on the phrases listed in Table 6.1 and the same word order as in its original translation.

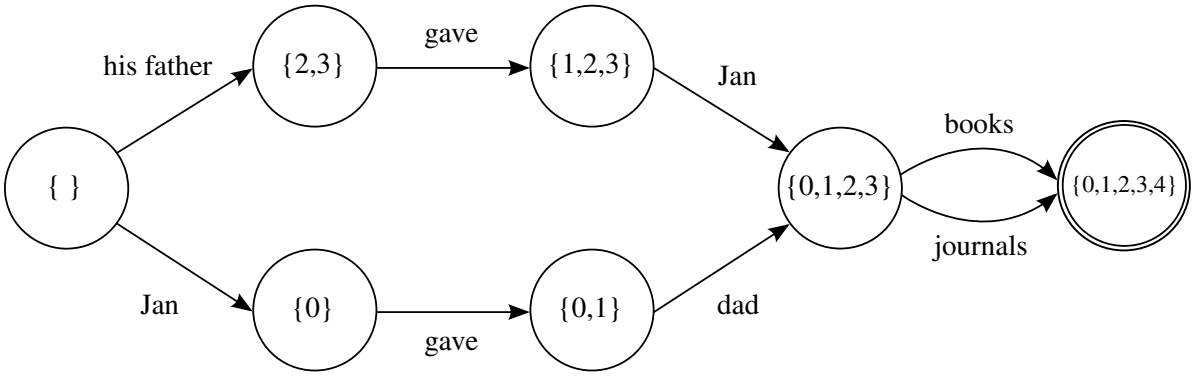

Figure 6.4: Lattice generation: translation 2 *Jan gave dad journals* has been inserted into the network based on its phrase alignments. New translation options already emerge as both translation share the same node $\{0,1,2,3\}$ and the phrases "books" and "journals" translate the same source span $\{4\}$.

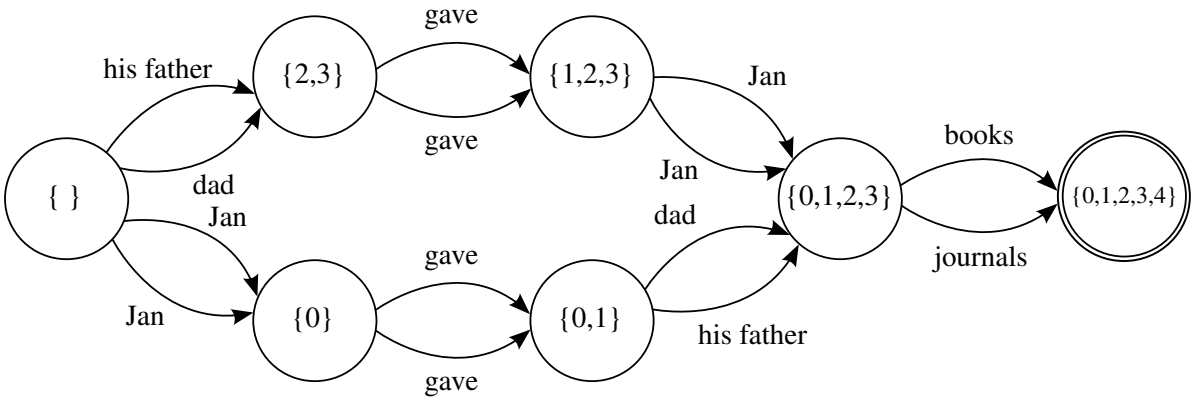

Figure 6.5: Lattice generation: adding phrases independent of their original position into the lattice only based on the phrase spans as listed in Table 6.1.

6.3.3 Hierarchical Phrases

Hierarchical phrase-based translation (cf. Section 3.6) is a widely used approach for translating language pairs containing long-range reorderings. Hierarchical phrase-based translation has the option to use additional to the original phrases, hierarchical phrases which are phrases containing sub phrases. Hierarchical phrases are problematic as it is impossible to order the phrases according to the positions

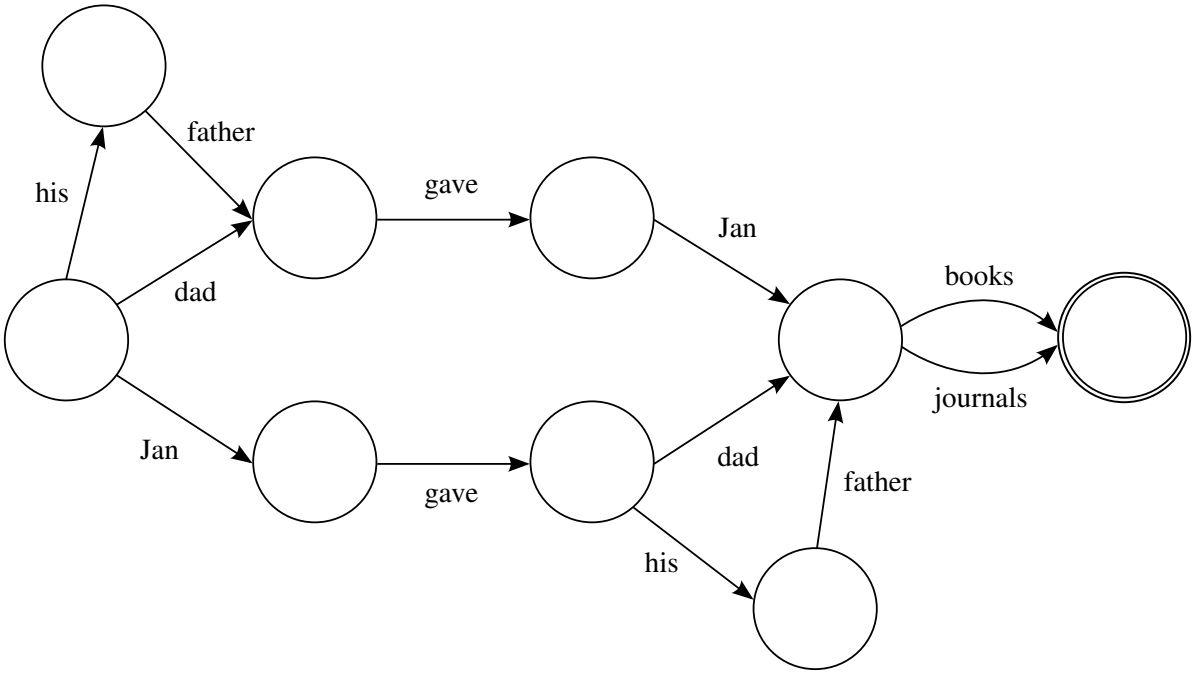

Figure 6.6: Lattice generation: all phrases of Figure 6.5 have been split into single words (for an arc labeled with two words, a new node is needed to be inserted). Finally, we make the lattice deterministic which includes merging arcs labeled with the same word at the same position.

in their translation. A hierarchical phrase with one gap is split into two normal phrases. Both new phrases only translate half of the source words. Instead of distributing the source positions between the two novel phrases, we assign each phrase the half of each source position. E.g. a hierarchical phrase AX_0B which is a translation of source positions i and j is split into two phrases A and B both translating positions $i/2$ and $j/2$. If word-to-word alignment information are given (which is mostly not the fact), we use the actual source words instead of $i/2$ and $j/2$ and distribute the source positions between the new phrases.

6.3.4 Models

Once we have the final lattice, we want to adopt models which are valuable models to score the different translation options. For fair comparison, we define a similar set of models compared to the target-based approach. The following set of models are used:

N **binary system voting models:** For each word the voting model for system n ($1 \leq n \leq N$) is 1 iff the phrase is from system n, otherwise 0.
m**-gram counts:** Four different m-gram counts (1-4 gram) calculated on the input hypotheses.
Word penalty: Counts the number of words.

6.3.5 Decoding

We obtain a combined hypothesis from the lattice in a similar manner as for the target-based system combination. We combine the M different models $h_m(e)$ in a linear framework and assign each model m its scaling factor λ_m. We then learn system weights for each model with MERT [Och 03]. Subsequently, we score the lattice with the learned system weights and each arc e gets assigned one score $S(e)$:

$$S(e) = \sum_{m=1}^{M} \lambda_m h_m(e) \tag{6.1}$$

Finally, we apply the shortest path algorithm [Mohri 02] to obtain the path with the lowest score whose labels are the combined translation. Although the lattice is no confusion network, the lattice is a directed cycle-free graph and the shortest path can easily be determined.

6.3.6 n-best System Combination

In traditional confusion network system combination we are able to add alternative translations (n-best lists) of each system output in addition to the confusion network. It has been common that only the first best outputs serve the word order and all alternatives are word-to-word aligned to the previous inserted first best translations as otherwise the confusion network is getting to large.

The novel approach is also able to add n-best lists into the lattice to generate additional translation alternatives. We add all phrases in the same manner as the first best translation to the lattice. As the alternative translations usually have small differences to the first best output, the lattice size only increases slightly. To give higher ranked hypotheses a higher influence, we modify the n-gram count calculation. The i-th entries of an n-best list only gives a count of $1/i$ to the n-gram counts.

6.4 Benefits of Source-Aligned Lattices

In this section, we investigate the differences between traditional confusion networks and the novel lattices produced with source information. In the following, we give various examples which demonstrate the benefit of using source phrase information during system combination. We compare the source-aligned lattice approach with two target-based confusion network approaches. For the target-based approach the alignments between the systems are learned either via GIZA++ or METEOR. All examples are real data examples and are extracted from the combined BOLT Chinese→English test set translations.

6.4.1 m-to-n Alignments

Instead of the usual learned word-to-word alignments, the source-aligned system combination is constructing m-to-n phrase alignments. In Example 1, nine individual system outputs are given. For a word-to-word system combination, it is impossible to align ”hard work” to ”diligence”, as it is an m-to-n alignment. In Figure 6.7, the confusion networks constructed by both the METEOR and the GIZA++ alignment are illustrated. Both fail to align the phrases, as both can only align one word to diligence. The source-aligned system combination instead uses information from the source phrases and thus detects the connection between ”hard work” and ”diligence”. In Figure 6.8 the source-aligned lattice is illustrated.

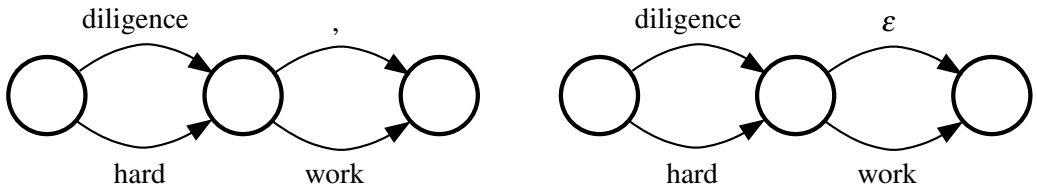

Figure 6.7: Target-based confusion network constructed with GIZA++ and METEOR alignments. The alignment procedure is unable to generate m-to-n alignments.

```
people need to unite , hard work
people need solidarity , diligence ,
people need unity , hard work
the people need solidarity , diligence ,
the people need solidarity , diligence ,
people need solidarity , diligence ,
people need unity , hard work
people need solidarity , hard work ,
people need solidarity , hard work ,
```

Example 1: Nine individual system outputs. The target-based alignment algorithms are unable to align ”hard work” to ”diligence”.

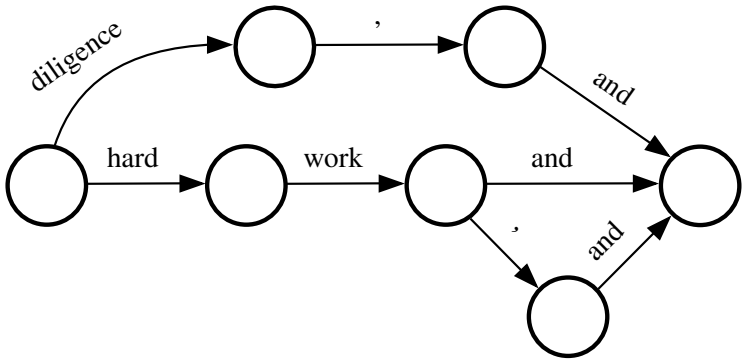

Figure 6.8: The lattice constructed with source alignment information gives ”hard work” as alternative for ”diligence” and vice versa.

6.4.2 Misalignments

It may happen that the traditional target-based approach fails to detect the connections between all alternative translations of the same source word. As a consequence, it is possible that the combined translation contains more than one translation of the same source word. With the help of the source information, we can guarantee that the combined output is a combination of translations from different systems, but each source word is only translated once by one system. Even if translations of one source word are identical, the word position in the individual translations can differ. If the word order is to different, the alignment approach can be confused and fails to align also identical translations. In Example 2, the word ”positive” is part of the translation of the Chinese word 正面‘ in eight of nine systems. Only one system does not produce it. The source alignment give us the information that all ”positive” are translations of the same source word and need to be aligned. However, without this information, we overproduce the word ”positive” in the final translation as it has been aligned twice to the phrases ”positive point” and ”positive achievements”. As illustrated in Example 2, the target-based system combination produce two positive which is wrong in case of the source information given by the individual systems.

6.4.3 Mixing Phrases

A phrase usually has more than one correct translation. The GIZA++ and METEOR alignment can not handle all alternative translations (also because of the word-to-word alignment restriction). In the Example 3 we have three different phrase translations ”the result is”, ”as a result” and ”as a result of”. The source-aligned system combination only produces one of the three options. The target-based

```
... a positive point of view the achievements of science
... the positive achievements in science
... positive achievements in the scientific circles
... the achievements of science
... a positive perspective of science  achievements
... a positive perspective the achievements of science
... the achievements of the industry from a positive angle
... the positive achievements in science
... the positive achievements of science

a positive point of view the achievements of science
a positive point of view the positive achievements of science
```

Example 2: Nine individual system outputs. The last two lines are system combination results: second last line is the combination result of a source-aligned lattice, last line is the result of a target-based system combination.

system combinations can in addition produce wrongly mixed outputs as happened in Example 3. The new resulting phrase ”as a result is” is a new constructed phrase. As marginal note, the new generated phrase can improve automatic scores, but most certainly decline the output for human judgements.

```
the result is ...
as a result ...
as a result of ...

the result is ...
as a result is ...
```

Example 3: First three lines are individual system outputs. Second last line is the source-aligned system combination output, last line is the target-based system combination output.

6.4.4 Alternative Translations

The source alignment information also improve the word-to-word alignment quality. Without any source information, we generate alignment errors due to wrong or missing alignment decisions. Obviously, the source information fix this problem. Additionally, we do not have any empty alignments. An empty alignment e.g. occurs in a target-based system combination, if a word in the actual hypothesis can not be aligned to the skeleton translation. This leads to empty translation segments and thus to empty translations of source words (even if that was not the case in any of the individual systems). Additionally, the phrase-to-phrase alignment is able to keep the fluency as it does not only rely on word selections.

6.4.5 Word Orders

An additional advantage of the source-aligned approach is that the lattice explores all word orders used in the hypotheses and not choose the word order of one skeleton or (when multiple skeletons are used) stick to the word order of one system throughout. With the source alignment information it is easy to switch at any phrase boundary to a word order of a different input system.

6.5 Experiments

All experiments are conducted with the system combination toolkit Jane [Freitag & Huck+ 14]. For both the source-aligned and the target-based system combination, we use the same decoding algorithm as well as the same models. The only difference between these approaches is the lattice construction. Translation quality is measured in lowercase with BLEU [Papineni & Roukos+ 02] and TER [Snover & Dorr+ 06] on single reference translations whereas the performance of each setup is the best score across five different MERT runs on the tune set. The system combination weights are optimized with MERT on 200-best lists with BLEU-TER as optimization criterion.

6.5.1 BOLT Chinese→English

We utilize nine individual systems to perform the BOLT Chinese→English experiments. All nine input systems are statistical machine translation engines with different extensions of either the usual phrase-based or the hierarchical phrase-based machine translation approach. Table 6.2 contains the empirical Chinese→English results. Comparing both target-based alignment approaches, the GIZA++ alignment performs 0.3 points in BLEU and 0.2 points in TER better. Including 5-best entries in the target-based system combination does not improve the translation quality. The source-aligned system combination enhances the translation quality by 0.5 points in BLEU and 0.4 points in TER. Adding the 5-best translations of each individual system to the source-aligned system combination further improves the translation quality by 0.6 points in BLEU and 0.5 points in TER compared to the GIZA++ target-based system combination.

Table 6.2: Experimental results on the BOLT Chinese→English data. All system combinations are conducted with nine different individual input systems. The novel source-aligned approach outperforms both target-based approaches in BLEU and TER.

system combination			tune		test	
method	**alignment**	**5-best list**	BLEU[%]	TER[%]	BLEU[%]	TER[%]
target-based (Chapter 4)	**METEOR**	**no**	20.1	61.5	18.1	61.1
	GIZA++	**no**	20.1	61.3	18.4	60.9
	GIZA++	**yes**	20.1	61.3	18.4	60.9
source-aligned		**no**	20.6	60.1	**18.9**	**60.5**
		yes	20.7	60.1	19.0	60.4

6.5.2 BOLT Arabic→English

Table 6.3 contains the empirical BOLT Arabic→English results. All results are system combinations of five individual systems which are statistical machine translation engines with different extensions of the usual phrase-based machine translation approach. The GIZA++ alignment outperforms the METEOR alignment with 0.1 points in BLEU and 0.3 points in TER. Similar to the BOLT Chinese→English task, the 5-best target-based system combination gives us no additional improvement. The source-aligned system combination improves the target-based system combination results in terms of 0.6 points in BLEU while losing 0.2 points in TER. When adding the 5-best translations to the source-aligned lattice, we achieve a total improvement of 0.8 points in BLEU while still losing 0.2 points in TER compared to the target-based system combination with GIZA++.

Table 6.3: Experimental results on the BOLT Arabic→English data. All system combination are conducted with five individual input systems. The novel source-aligned system combination outperforms both target-based confusion network approaches.

system combination			tune		test	
method	**alignment**	**5-best list**	BLEU[%]	TER[%]	BLEU[%]	TER[%]
target-based (Chapter 4)	**METEOR**	**no**	27.0	55.1	27.5	56.1
	GIZA++	**no**	27.0	54.7	27.6	55.8
	GIZA++	**yes**	27.0	54.7	27.6	55.8
source-aligned		**no**	27.5	55.3	28.2	56.0
		yes	27.4	55.4	28.4	56.0

6.6 Human Evaluation

In this section, we compare the network quality of the source-aligned networks with the target-based confusion networks generated by either GIZA++ or METEOR. We perform human evaluation of the first 30 sentences of the given BOLT Chinese→English test set. In the following section, we introduce word alignment error classes for both the target-based and the source-aligned lattices. We need to define different error classes for both the source-aligned and target-based networks, because their structures are different and produce different kinds of errors.

6.6.1 Error Classes (Target-Aligned)

We divide the error classes for the traditional target-based confusion networks into two main categories, which are missing alignment and wrong alignment. A missing alignment error refers to a word which should be aligned but is not. A wrong alignment error refers to a word which is aligned, but should not. The two main categories are further subdivided into six sub categories:

- missing alignments:
  - 1-*n* alignment
  - same meaning
  - same source word
  - same word
- wrong alignments:
  - wrong meaning
  - wrong position

1-*n* alignment: Confusion networks are based on word-to-word alignments and thus are unable to align 1-to-*n* phrase alternatives. In Figure 6.9, one 1-*n* alignment error example is illustrated. The alignment algorithm is unable to align the two words ”30 thousand” to the single word ”30,000”. It is possible that the final translation contains both translations or neither of them.

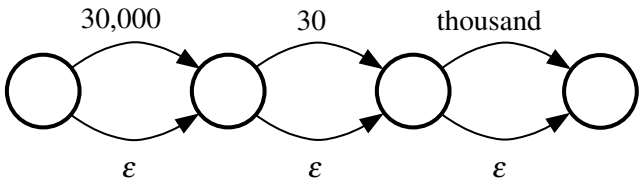

Figure 6.9: Example of a 1-*n* alignment error.

Same meaning: We call a missing alignment error ”same meaning”, if two different translations of the same source word have the same meaning, but are not aligned. An example is illustrated in Figure 6.10: the two translations "several" and "repeated" should be aligned as they are alternative translations of the same source word.

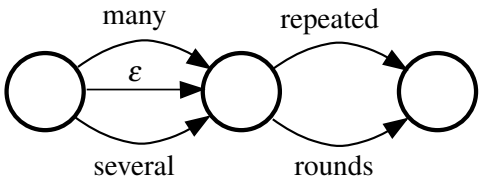

Figure 6.10: Example of a same meaning error.

Same source word: Different translations of the same source word need to be aligned, even if their meanings are different. This kind of errors are called ”same source” error. In Figure 6.11, the words ”repair” and ”distribution” are translations of the same source word and should be aligned, even though they have different meanings.

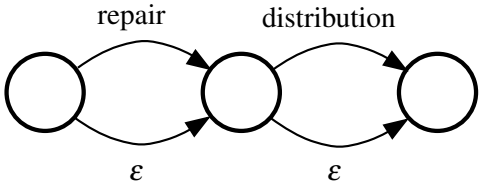

Figure 6.11: Example of a same source word error.

Same word: A same word error occurs, if two individual systems generate identical translations for the same source word and these are not aligned. In Figure 6.12, the word ”revealed” is produced by different individual systems, but is not aligned.

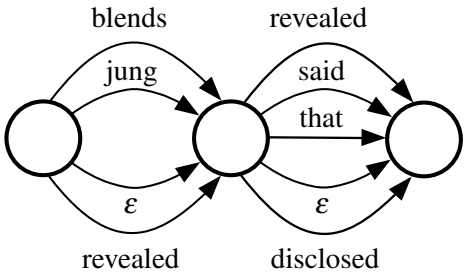

Figure 6.12: Example of a same word error.

Wrong meaning: A wrong alignment is categorized as a ”wrong meaning” error, if the translations of different source words are aligned and additionally the translations have different meanings. In Figure 6.13, the words ”scholar” and ”time” are wrongly aligned.

Wrong position: Translations of different source words should not be aligned, even if they have the same meaning. A wrong alignment is called ”wrong position” error, if two synonyms are aligned, but translations of two different source words. In Figure 6.14, the words "builds" and "building" are wrongly aligned as they are translations of different source words. This kind of wrong alignment is critical as they yield to untranslated source words.

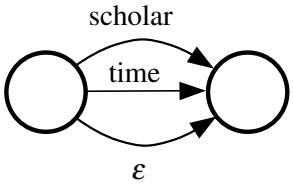

Figure 6.13: Example of a wrong meaning error.

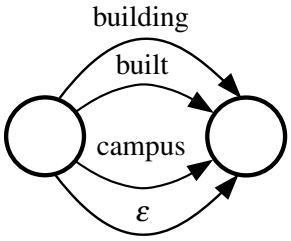

Figure 6.14: Example of a wrong position error.

6.6.2 Error classes (Source-Aligned)

We define two error classes for the source-aligned lattices:

- duplicate alignment
- wrong alignment

Duplicate alignment: Identical translations can be misaligned as the source alignment information can be different. In Figure 6.15, all ”the” should be aligned as they are identical translations of the same source word. We categorize an alignment error as ”duplicate alignment”, if identical translations of the same source word are not aligned.

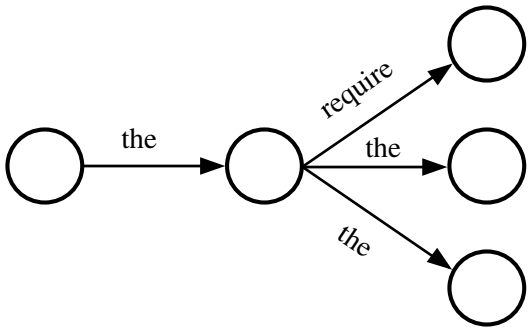

Figure 6.15: Example of a duplicate alignment error.

Wrong alignment: Due to wrong phrase alignments, different translations of different source words can be aligned. This error obviously occurs mostly at phrase boundaries and is a result of the phrase extraction algorithm in phrase-based and hierarchical phrase-based machine translation. In Figure 6.16 an example of a ”wrong alignment” error is given: ”capital” and ”fund” are wrongly aligned as they are translations of different source words.

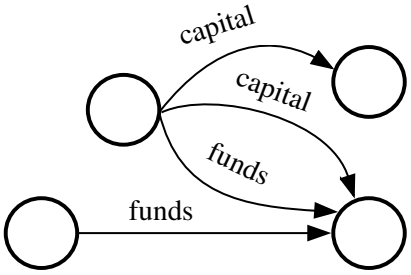

Figure 6.16: Example of a wrong alignment error.

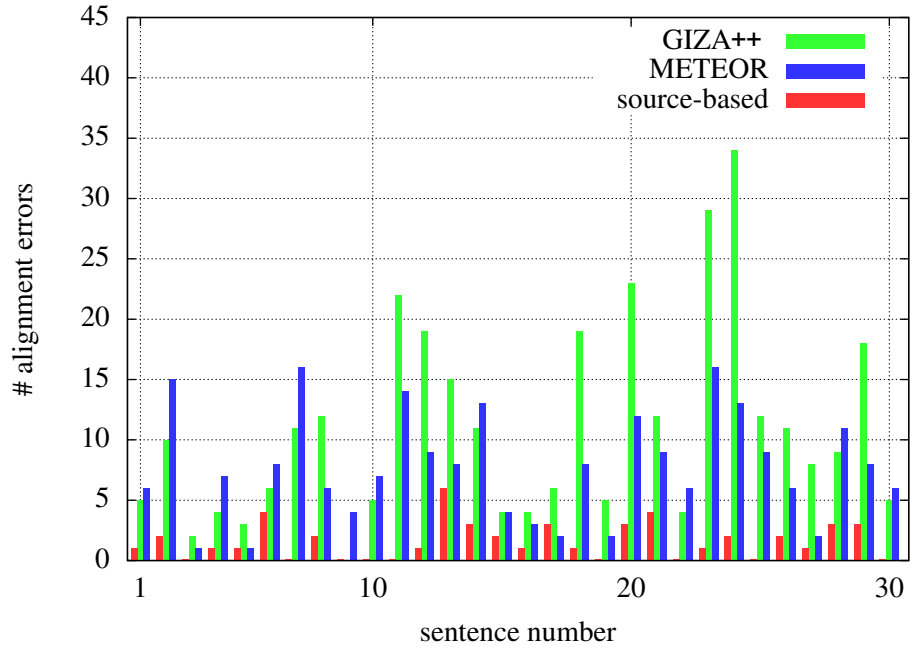

Figure 6.17: Error analysis of the first 30 test set sentences for the BOLT Chinese→English translation task: the novel source-aligned alignment approach outperforms the traditional alignment approaches generated by either GIZA++ or METEOR. The confusion networks generated by METEOR produce less alignment errors compared to the confusion networks generated by GIZA++.

6.6.3 Alignment Error Statistics

In this section, we evaluate the different network approaches on the first 30 sentences of the test set of the BOLT Chinese→English translation task. The human evaluation results are illustrated in Figure 6.17. We compare the source-aligned lattices with both target-based confusion networks generated by either GIZA++ or METEOR. The source-aligned lattice outperforms both target-based lattices for all 30 sentences. Furthermore, the novel source-aligned lattices do not contain any error in 9 of the 30 sentences. In contrast to the translation results presented in Table 6.2, the lattices generated by METEOR have a lower error count compared to the lattices generated by GIZA++.

The sum over all 30 sentences of the different error categorizes for both target-based confusion networks are illustrated in Figure 6.18. GIZA++ produces more wrong alignments whereas the METEOR algorithm misses more alignments. The METEOR alignment only aligns words, if they are synonyms or have the same stem. Contrary to the METEOR alignment, the GIZA++ alignment is able to capture translation alternatives which are no synonyms or share the same stem. This results to the

fact that GIZA++ produces more unsure alignments. Nevertheless, the GIZA++ alignment approach yields better translation results as seen in Section 6.5.

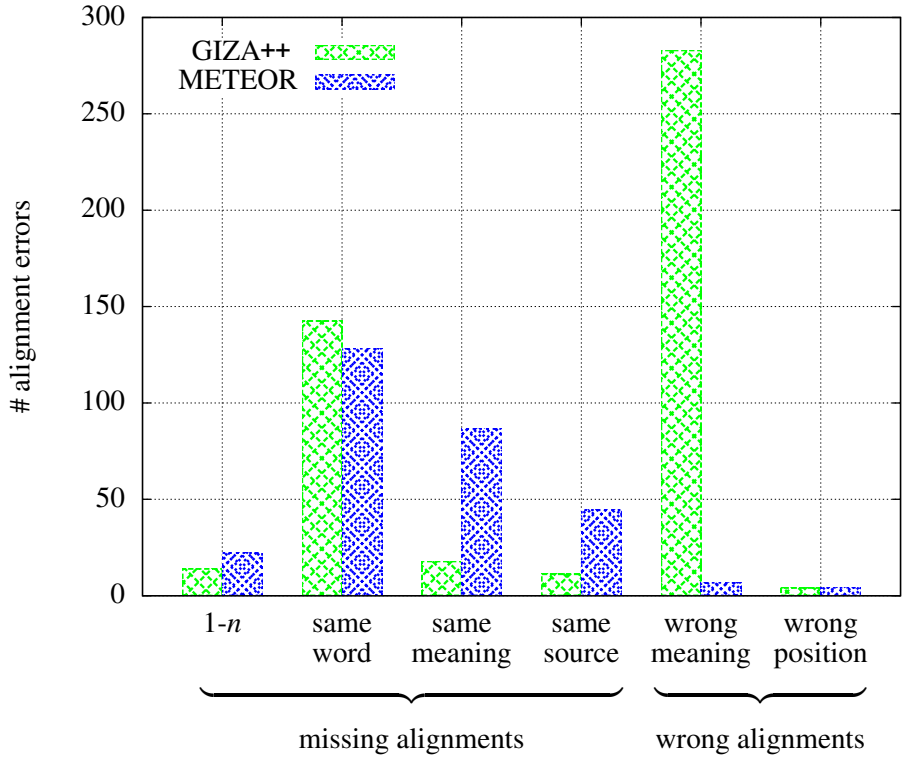

Figure 6.18: Error analysis of the first 30 test set sentences for the BOLT Chinese→English translation task: the GIZA++ alignment approach generates more wrong alignments, but also misses less correct alignments compared to the METEOR alignment approach.

The two different error categorizes of the source-aligned lattices are compared in Table 6.4. The source-aligned lattice approach produces the most errors in generating duplicate alignments which occur due to wrong phrase alignments. This kind of error should be easily captured by the language model (trained on the input hypotheses) which rarely gives word sequences of repeating words a change to be in the final output. Nevertheless, we prevent the generation of new output by setting the weight of the language model to high as it prefers the word sequences seen in the individual system themselves.

Table 6.4: Error analysis of the first 30 test set sentences for the BOLT Chinese→English translation task: the novel source-aligned alignment approach produces more duplicate alignment errors than wrong alignment errors.

error category	# alignment errors
duplicate alignment	39
wrong alignment	8

6.7 Translation Examples

In this section, we present three translation examples which show the improvement of lattice decoding by using the source alignment information. All examples are extracted from the BOLT Chinese→English setup. We compare the target-based GIZA++ and source-aligned setups. All scores are measured with TER and SBLEU. To allow for sentence-wise evaluation, [Lin & Och 04] define the SBLEU metric which is basically BLEU where all n-gram counts are initialized with 1 instead of 0. The brevity penalty is calculated only on the current hypothesis and reference sentence.

In Table 6.5, the first translation example is illustrated. The target-based confusion network decoding approach generates one new mixed phrase ”from this we can be seen” which has been built from the phrases ”from this we can see” and ”it can be seen”. This obviously wrong new phrase can not be built by the source-aligned lattice approach, as only one of these two options is allowed due to the source alignment restriction. In the second example (Table 6.6), the phrase ”is good” has been skipped by the target-based confusion network decoding approach. In the last example illustrated in Table 6.7, the target-based confusion network decoding skips the translation of several source words. This can happen if words can not be aligned and thus are aligned to an ε token. If this occurs for many translations of the same source word, this source word is untranslated. In this example, the target phrase ”quiet and” is ignored which is not possible for the source-aligned system combination. The source alignment information guarantees to translate each source word exactly one time.

Table 6.5: Translation examples extracted from the BOLT Chinese→English translation task. We compare the target-based GIZA++ confusion network approach with the novel source-aligned lattice approach. The source alignment information helps system combination to translate each source word exactly once: The translations ”from this we can see” and ”it can be seen” of the same source segment have been mixed in the target-based approach. The translation scores are: TER: 50.00 SBLEU: 38.12 (target-based), TER: 40.00 SBLEU: 41.34 (source-aligned).

reference	from this you can see just how much public credibility of this kind of quality sample reports can have .
target-based	from this we can be seen that , how much can the credibility of this kind of quality inspection report .
source-aligned	from this we can see that , how much can the credibility of this kind of quality inspection report .

Table 6.6: Translation examples extracted from the BOLT Chinese→English translation task. We compare the target-based GIZA++ confusion network approach with the novel source-aligned lattice approach. The translations ”is good” has been skipped by the target-based approach. The source alignment information guarantees that each source is translated. The translation scores are: TER: 50.00 SBLEU: 40.19 (target-based), TER: 25.00 SBLEU: 76.75 (source-aligned).

reference	do you say this society is good ?
target-based	you say that this society ?
source-aligned	you say that this society is good

Table 6.7: Translation examples extracted from the BOLT Chinese→English translation task. We compare the target-based GIZA++ confusion network approach with the novel source-aligned lattice approach. The source alignment information helps system combination to translate each source word exactly once: E.g. the translation "quiet and" has been skipped by the target-based approach. The translation scores are: TER: 58.82 sBLEU: 26.04 (target-based), TER: 47.06 sBLEU: 35.91 (source-aligned).

reference	Chinese are too quiet and reserved , rui chenggang has given strength , i support him .
target-based	the Chinese people are too subservient , rui chenggang awesome , support .
source-aligned	the Chinese people are too quiet and subservient , rui chenggang awesome , and support .

6.8 Conclusion

In this chapter, we introduced a novel source-aligned phrase-to-phrase lattice system combination approach. We used the phrase information of all individual systems to align the different machine translation systems. By doing so, we solved many alignment errors which occur in a target-based system combination as wrong word repetitions, lack of *m*-to-*n* alignments, empty translations, multiple translations of the same source sequence or simple word-to-word alignment errors. Additional to that, the phrase information gives us the potential to insert *k*-best list alternative phrases which increases the translation quality. The final translation output of the proposed lattice decoding approach does not only improve the automatic metrics, but also augments the fluency of the output. We defined error classes for the different lattice types and came to the conclusion that the source-aligned lattices produce fewer errors compared to the target-based networks. Experiments show translation quality improvement by 0.6 points in BLEU and 0.5 points in TER. Unfortunately, it is not possible to integrate the localVote feature of the previous chapter into the lattice system combination approach. The localVote feature is based on a word-to-word lattice which let you directly choose between the words from the different input hypotheses. This is essential for the training of the neural network model and is no longer given in our novel approach.

7 Reverse Word Order Models

In this chapter, we study the impact of the word order decoding direction for statistical machine translation. Both phrase-based and hierarchical phrase-based machine translation approaches are investigated by reversing the word order of the source and/or target language and comparing the translation results with the normal direction. We analyze alignment model, language model, and phrase table extraction to investigate the effect of reversing the individual components. Furthermore, we propose to use system combinations, alignment combinations, and phrase table combinations to take benefit from systems trained with different translation directions. Experimental results show improvements of 1.7 points in BLEU and 3.1 points in TER for the NTCIR-9 patent Japanese→English and Chinese→English tasks and 1.0 point in BLEU and 0.9 points in TER for the BOLT Chinese→English translation task compared to the normal direction systems.

7.1 Introduction

The decoding direction of a phrase-based statistical machine translation engine affects the resulting translation and can be a critical point for the translation quality. Furthermore, both the language model and the alignment training yield different results when estimating the models on reversed sentences. In this chapter, we reverse the word order from left-to-right to right-to-left for source and target sentences to produce a reversed bilingual corpus. The decision whether to use a reversed word order can be taken individually for each the alignment training, the language model training, and the decoding process. We build fully reversed systems as well as systems for which only parts are trained on reversed corpora. Furthermore, we train systems based on corpora with just source or target language reversed. We analyze which methods depend on the word order and which one can benefit from a reversed word order. We run alignment combinations, phrase table combinations, and system combinations of up to eight normal and reverse systems to show the improvement of combining the benefits of both translation directions. To make the comparison fair, the normal and reversed systems have the same system setups, i.e. only the word order varies. Most of the work described in this chapter has been published in [Freitag & Feng$^+$ 13].

This chapter is structured as follows. In Section 7.2, we give an outline of the related work. The reverse translation approaches and the combination algorithms are described in Section 7.3 and Sec-

tion 7.4. We analyze the differences of the alignment model and language model training as well as the differences of the decoding process in Section 7.5. The experimental results are presented in Section 7.6. Finally, we discuss the results in Section 7.8.

7.2 Related Work

[Watanabe & Sumita 02] describe a right-to-left decoding method for a standard phrase-based machine translation decoder. In addition, the authors introduce a bidirectional decoding method, which combines the advantages of both left-to-right and right-to-left decoding methods by generating the output in both ways. Experimental results show that the right-to-left decoding is better for English-to-Japanese translation, while the left-to-right decoding is suitable for Japanese-to-English translation. They also show that the bidirectional method yields the best translation performance for the English-to-Japanese translation task. The authors suggest that the translation output generation should match the underlying linguistic structure of the output language.

[Finch & Sumita 09] compare a standard phrase-based machine translation decoder using a left-to-right decoding strategy to a right-to-left decoder for several language pairs on small training corpora. The authors demonstrate that for most language pairs, right-to-left decoding yields better performance than left-to-right decoding. However, the performances of left-to-right and right-to-left strategies seem to be highly language dependent. The word order of the target language partially accounts for the differences in performance when decoding in different directions.

[Xiong & Zhang$^+$ 11, Xiong & Zhang 15] train a backward language model which assigns a score to the succeeding n-1 words plus the current word. They train a backward n-gram language model on reverse corpora and integrate the forward and backward language models together into the decoder. In doing so, they attempt to capture both the preceding and succeeding contexts of the current word. They show improvements from up to 0.5 points in BLEU by using both language models together. In 2015, they published an update of their work which in addition compares the backward language model to the forward language model. They come to the conclusion that the backward language model is competitive with the forward language model and significantly outperforms the forward language model in Chinese→English and Vietnamese→English translation.

[Duchateau & Demuynck$^+$ 02] introduce the backward n-gram language model as an additional information source for confidence measures in speech recognition. Experiments on the WSJ recognition task show that the backward language model contains information that is complementary to the information in the forward language model. Adding this information to the confidence measure (which is also based on the forward language model) results in an increase of the normalized cross entropy from 18.5% to 23.3%.

[Frinken & Fornés$^+$ 12] In order to improve the results of automatically recognized handwritten text, a language model is commonly included in the recognition process. The authors propose to generate two different n-best lists of recognition hypotheses, one generated by a left-to-right and one generated by a right-to-left decoding. Afterwards, these n-best lists can be combined using a generalized recognizer output voting error reduction (ROVER) [Fiscus 97] scheme. The experimental results obtained with the ROVER combination have shown a significant improvement over current state-of-the-art approaches.

In the first two machine translation papers [Watanabe & Sumita 02, Finch & Sumita 09], the authors only change the decoding direction while using the same alignment. In addition to the decoding direction, we also investigate the impact of a right-to-left alignment training. Instead of a bidirectional decoding method, we use alignment combinations, phrase table combinations, and system combina-

tions to benefit from both translation directions. In addition to the phrase-based decoder, we are also using a hierarchical phrase-based decoder in all our experiments. Further, we train translation systems based on partially reversed corpora (reversing either the source or target side only).

[Xiong & Zhang$^+$ 11] learn a backward language model which takes the succeeding words into account. They integrate both language models into the decoder and in doing so they take the benefits of both directions into account. We only use either one forward language model or one backward language model for all our systems. [Duchateau & Demuynck$^+$ 02] use the backward language model in a similar way as [Xiong & Zhang$^+$ 11], but for confidence measure for speech recognition.

Similar to our work, [Frinken & Fornés$^+$ 12] train a complete reverse system for handwriting recognition and combine the two system with system combination. Different to machine translation, the handwriting recognition does not need an alignment step. The author mainly take the benefits of the reversed corpora from the two different trained language models as well as from the slightly different decoding schemes.

In summary, the novel contributions of this work and the differences to the above publications are:

1. Retraining of the alignment model with reversed corpora.
2. Usage of both hierarchical phrase-based and phrase-based decoders.
3. Building translation systems based on partially reversed corpora.
4. Application of alignment combinations, phrase table combinations, and system combinations.
5. Evaluation on large-scale tasks and data of recent public evaluation campaigns.

7.3 Reversed Corpora

Instead of adapting the decoding algorithm and decode from right-to-left, we only change the word order of the bilingual corpus and simulate right-to-left decoding. For example, if we reverse both source and target sentences, the original training example "der Hund mag die Katze . $\rightarrow$ the dog likes the cat ." is converted into a new training example ". Katze die mag Hund der $\rightarrow$. cat the likes dog the". We denote this type of modification of source or target sentences by the term *reversion*. A system trained of such data is called *reversed*. This modification changes the training corpus, hence the language model and alignment training produce different results.

In the following, various reversed and partial reversed systems are defined. For a source sentence $f_1^J = f_1, ..., f_J$ and a target sentence $e_1^I = e_1, ..., e_I$ we define the following systems:

- **normal system**:
  - normal corpus: $f_1, f_2, ..., f_J$ and $e_1, e_2, ..., e_I$
  - alignment, language model, and phrase table are trained on a corpus with original word order.
- **reversed system**:
  - reversed corpora: $f_J, f_{J-1}, ..., f_1$ and $e_I, e_{J-1}, ..., e_1$
  - alignment, language model, and phrase table are trained on a corpus with reversed word order.
- **source-reversed system**:
  - source-reversed corpora: $f_J, f_{J-1}, ..., f_1$ and $e_1, e_2, ..., e_I$
  - alignment, language model, and phrase table are trained on a corpus with reversed source word order and original target word order.
- **target-reversed system**:
  - target-reversed corpora: $f_1, f_2, ..., f_J$ and $e_I, e_{I-1}, ..., e_1$
  - alignment, language model, and phrase table are trained on a corpus with reversed target word order and original source word order.
- **alignment-reversed system**:
  - normal corpus: $f_1, f_2, ..., f_J$ and $e_1, e_2, ..., e_I$ for decoding
  - alignment is trained on a corpus with reversed word order (as in a *reversed system*).
  - language model and phrase table are trained on a corpus with original word order (as in a *normal system*).
- **lm-reversed system**:
  - reverse corpus: $f_J, f_{J-1}, ..., f_1$ and $e_I, e_{J-1}, ..., e_1$ for decoding
  - language model is trained on a corpus with reversed word order (as in a *reversed system*).
  - alignment and phrase table are trained on a corpus with original word order (as in a *normal system*).

7.4 Alignment Combination and Phrase Table Combination

Alignment combination: A two-directional alignment training is performed by combining the final normal and reversed alignments. We train two alignments, i.e. one on a corpus with normal word order and the other one on a corpus with a reversed word order. We merge the two resulting alignments with one of the symmetrization heuristics grow-diag-final (*gdf*), *iu*, *intersection*, or *union* as described in [Och & Ney 03].

Phrase table combination: A two-directional phrase table is a combination of two phrase tables. We use two different methods for combining phrases. *Intersection* only keeps phrases which exist in both phrase tables. The model scores are the average of the model scores of both phrase tables. *Union* is a superset of intersection. In addition to the resulting phrases of the intersection algorithm, we keep the phrases which only exist in one of the two phrase tables.

The combination method *intersection + 4 models* is based on intersection. We keep the same set of phrases as for the *Intersection* algorithm. Instead of taking the average of both phrase translation probabilities and both lexical smoothing probabilities, we just keep all of them in the phrase table. The definition for *union + 4 models* is equivalent to the definition of *intersection + 4 models* while only changing the combination method.

For both the phrase table and alignment combination, we first reverse the reversed trained alignment as well as the reversed trained phrase table to the normal word order.

7.5 Analysis of Reversed and Normal Systems

In the following section, we point out the differences between a reversed and normal system. We analyze the alignment and language model training. For both the phrase-based and hierarchical phrase-based decoder, we analyze the phrase extraction and the decoding process.

The phrase-based translation (PBT) systems use the standard set of models including phrase translation and lexical smoothing probabilities in both translation directions, word and phrase penalty, a distance-based distortion model, an 4-gram target language model, and four binary count models. For our hierarchical phrase-based (HPBT) setups, we used phrase translation probabilities and lexical smoothing in both translation directions, word and phrase penalty, binary models marking hierarchical phrases, glue rule, and rules with non-terminals at the boundaries, four binary count models and an 4-gram language model.

All our analyses are conducted on both NTCIR-9 PatentMT[1] tasks, namely Japanese→English and Chinese→English. Table 10.7 and Table 10.8 show the corpus statistics of the bilingual data used for the NTCIR-9 tasks. The language models are trained on the target side of the bilingual data and the monolingual data sets provided by the organizers.

7.5.1 Language Model

Language models are trained with the SRILM toolkit [Stolcke 02] using modified Kneser-Ney discounting. We compare a language model trained on a reversed corpus versus a language model trained on the original word order. The training for both language models is the same in terms of smoothing and amount of training data. Instead of taking the preceding n-1 words into account, a language model trained on a reversed corpus considers the succeeding n-1 words.

Table 7.1: Language model perplexities, all language models are 4-grams. Perplexity is a measurement of how well a probability distribution or probability model predicts a given test set. The two different approaches learn models with similar perplexity for both NTCIR-9 language pairs.

Language model	**Perplexity**
Chinese→English standard	59.1
Chinese→English reversed	58.9
Japanese→English standard	37.8
Japanese→English reversed	37.8

[1] http://research.nii.ac.jp/ntcir/ntcir-9/index.html

The perplexity information is given in Table 7.1. Both language models achieve similar numbers. Nevertheless, as both language models consider different context during decoding, the language model is still a crucial component while decoding from right-to-left instead of left-to-right.

7.5.2 Alignment

For the alignment training we utilize IBM model 1 (IBM-1) [Brown & Della Pietra$^+$ 93], HMM [Vogel & Ney$^+$ 96], and IBM model 4 (IBM-4) [Brown & Della Pietra$^+$ 93]. The word order does not affect the alignment probability if we use IBM model 1. Hence, for the reversed and normal systems IBM-1 produces the same result. However, the HMM model depends on the previous word and the training results in different alignments when conducted on reversed order corpora. Further, as previously described in Section 3.2.2, the probabilities of IBM Model 2 highly depend on the word order of the source and target sentences. IBM model 4 is an extension of IBM model 2 and it models also the probability of an alignment point depending on the source and target positions it connects. The absolute difference of the source and target positions can differ when counting word positions from the end to the beginning of a sentence (which simulates a reversed corpus). In general, the alignment training prefers a monotone alignment and gives alignment points for which the difference of source and target position is relatively small a higher probability.

In Figure 7.1 an example alignment for a normal system and a reversed system is illustrated. It can be observed that the alignments differ only slightly. The reversed trained alignment prefers the diagonal line starting from the upper right-hand corner, whereas the normal alignment prefers the diagonal line starting from the lower left-hand corner. As we are going to show later, the best choice is to use a combination of both alignments.

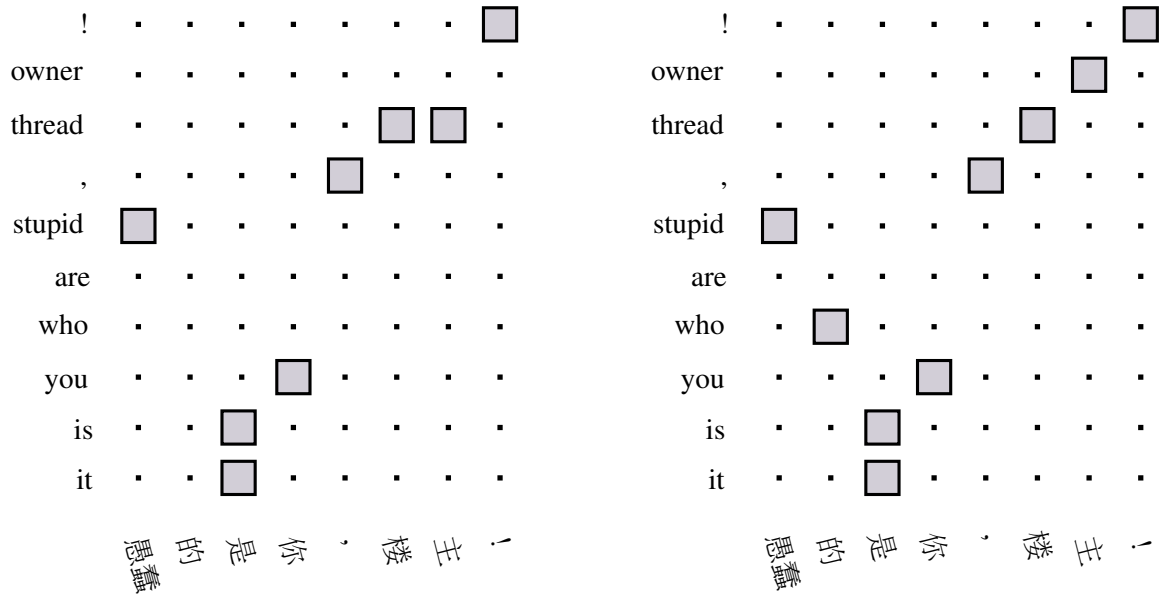

Figure 7.1: Example of a normal (left-hand side) and a reversed (right-hand side) trained alignment extracted from the BOLT Chinese→English data. Both alignments prefer the diagonal line which results to different alignments.

7.5.3 Phrase Extraction and Decoding

The phrase extraction settings for lexical as well as for hierarchical phrases are the same for the normal and the reverse direction. Formally, for a given sentence pair (f_1^J, e_1^I) with alignment A we extract all lexical bilingual phrases $BP(f_1^J, e_1^I, A)$ with following criterion:

$$BP(f_1^J, e_1^I, A) = \Big\{ (f_{j_1}^{j_2}, e_{i_1}^{i_2}) : \forall (j,i) \in A : j_1 \leq j \leq j_2 \leftrightarrow i_1 \leq i \leq i_2 \\ \wedge \exists (j,i) \in A : j_1 \leq j \leq j_2 \wedge i_1 \leq i \leq i_2 \Big\} \tag{7.1}$$

Followed from this equation, the source and target word order direction is not relevant for the lexical phrases. The hierarchical phrases are built from the lexical phrases. All heuristics for generating these are independent from the phrase direction. To verify this with our hierarchical phrase extraction, we built an *alignment-reversed* system and compare it with a normal system. The *alignment-reversed* system uses the same alignment as the normal system, but learns the phrases table on reversed sentences. In Table 7.2 the number of lexical as well as the number of hierarchical phrases are listed.

Table 7.2: Phrase table sizes for the NTCIR-9 Japanese→English subtask. The phrase extraction of both PBT normal and HPBT normal setups produce more phrases compared to the reversed phrase extractions. HPBT reversed and HPBT alignment-reversed produce the exact same amount of phrases.

Phrases	**HPBT normal**	**HPBT reversed**	**HPBT alignment-reversed**	**PBT normal**	**PBT reversed**
hierarchical	34 205 034	33 150 034	33 150 034	-	-
lexical	31 345 790	31 137 318	31 137 318	27 620 220	27 433 584
total size	65 550 824	64 287 352	64 287 352	27 620 220	27 433 584

We further study the number of unaligned words of the normal and the reversed system. In Table 7.4 and Table 7.5 the number of unaligned words are listed. The normal system has less aligned words than the reversed system. This number explains the fact that the normal phrase table size is larger than the reversed one, as shown in Table 7.2 and Table 7.3.

Table 7.3: Phrase table sizes for the NTCIR-9 Chinese→English subtask. As already seen for the Japanese→English translation task, the phrase extraction conducted on the normal word order sentences results in more phrases.

Phrases	**HPBT normal**	**HPBT reversed**	**HPBT alignment-reversed**	**PBT normal**	**PBT reversed**
hierarchical	20 303 097	19 883 111	19 883 111	-	-
lexical	14 473 801	14 464 621	14 464 621	14 427 127	14 354 721
total size	34 776 898	34 347 732	34 347 732	14 427 127	14 354 721

The decoding step of the hierarchical phrase-based system is independent of the word order direction. As the final translation is build hierarchical, it is irrelevant if the word order of the corpus is

Table 7.4: Amount of unaligned words for the NTCIR-9 Japanese→English subtask. The reversed alignment training produces less alignment points.

System	Source	Target
reversed	17 074 676	20 278 170
normal	18 039 699	20 598 882
total words	109 064 806	109 920 763

Table 7.5: Amount of unaligned words for the NTCIR-9 Chinese→English subtask. The alignment training conducted on the reversed word order sentences results in less alignment points.

System	Source	Target
reversed	4 787 877	7 129 962
normal	4 874 689	7 233 732
total words	41 249 103	42 651 202

reversed or normal. The hierarchical decoder proceeds with the search process in the same way for both directions.

For the standard phrase based approach, the search is done by Dynamic Programming Beam Search [Zens & Ney 08]. As seen in the publications mentioned in Section 7.2, the Dynamic Programming Beam Search gives different results when changing the decoding direction.

7.6 Experiments

We run various experiments with translation systems trained on different word order corpora as well as combinations of them. We build systems on both NTCIR-9 Japanese→English and Chinese→English corpora and on both BOLT Chinese→English and Arabic→English data sets. Each setup is run five times and the result is the best performing system based only on the development set. All experiments are evaluated with BLEU [Papineni & Roukos$^+$ 02] and TER [Snover & Dorr$^+$ 06].

7.6.1 NTCIR-9 Japanese→English

For the Japanese→English corpus, all experimental results are listed in Table 7.6. First, we compare for both translation systems the performance of a normal and a reversed system. For our standard phrase-based translation system, the reversed system performs better than the normal system by 0.7 points in BLEU and 0.8 points in TER. For our hierarchical system, the reversed system outperforms the normal system by 0.7 points in BLEU and 1.9 points in TER.

We first conduct a system combination with only the normal PBT and normal HPBT systems. We yield an improvement of 0.2 points in BLEU compared to the HPBT normal system. Nevertheless, the TER score is 0.3 points worse compared to the normal PBT system. In summary, system combinations that only uses the PBT normal and HPBT normal systems give no improvement.

Secondly, we combine the four hypotheses PBT normal, PBT reversed , HPBT normal, and HPBT reversed. We achieve an improvement of 1.7 points in BLEU and 3.1 points in TER compared to the

Table 7.6: Experimental results for the NTCIR-9 Japanese→English subtask. For all reversed systems, source and target language is reversed. For the source-reversed and target-reversed systems, only one language is reversed.

system	syscomb setup			dev		test	
	#1	#2	#3	BLEU[%]	TER[%]	BLEU[%]	TER[%]
PBT normal	yes	yes	yes	27.9	63.5	30.1	61.9
HPBT normal	yes	yes	yes	29.1	64.7	30.7	63.9
PBT reversed	-	yes	yes	28.9	62.9	30.8	61.1
HPBT reversed		yes	yes	29.6	63.3	31.4	62.0
HPBT alignment-reversed	-	-	-	29.4	63.1	31.3	61.9
HPBT lm-reversed	-	-	-	29.0	64.4	30.7	64.0
HPBT source-reversed	-	-	yes	28.0	64.1	30.0	62.4
HPBT target-reversed			yes	27.9	65.5	29.2	64.3
system combination setup #1				29.4	63.2	30.9	62.2
system combination setup #2	-	-	-	30.6	60.4	**33.1**	58.9
system combination setup #3				30.8	59.4	33.1	**58.1**

best single system HPBT reversed.

Thirdly, we use the hierarchical phrase-based approach to run some experiments with only reversing the source (source-reversed system) or the target corpus (target-reversed system). For the test set, the source-reversed system performs worse in BLEU compared to both HPBT normal and HPBT reversed systems. For TER, we achieve an improvement compared to HPBT normal, but also lose performance compared to HPBT reversed. The target-reversed system performs worse compared to both HPBT systems. Nevertheless, if we add the hypotheses to our system combination we achieve an additional improvement of 0.8 points in TER compared to the system combination of the first four systems in Table 7.6.

7.6.2 NTCIR-9 Chinese→English

In addition to reversed and normal systems, we also run experiments of combining alignments and phrase tables for the NTCIR-9 Chinese→English subtask. The results are listed in Table 7.7. For PBT, the normal system is slightly better than the reversed system with 0.3 points in BLEU. For HPBT, the reversed system performs better with 0.5 points in BLEU and 1.7 points in TER compared to the normal HPBT system. The HPBT alignment-reversed system performs similar to the HPBT reversed system. As the reversed word order does not affect the decoding process, the only difference of these systems is the language model as both systems are based on the same alignment and the decoding scheme of a hierarchical phrase-based decoder is independent of the word order. Based on our experiments, changing the forward language model to a backward language model (trained on reversed corpora) does not change the system performance for the NTCIR-9 Chinese→English task.

The system combination of the four systems *PBT normal*, *PBT reversed*, *HPBT normal*, and *HPBT reversed* gives us an additional improvement of 0.4 points in BLEU and 0.5 points in TER compared to our best single system. For the combination of the normal and reversed alignments, the best result is given by the union heuristic. We yield an improvement of 0.2 points in BLEU, but we lose 0.7 points

Table 7.7: Results for NTCIR-9 Chinese→English translation subtask. For all reversed systems, source and target language is reversed.

system	combination algorithm	syscomb setup #1	dev		test	
			BLEU [%]	TER [%]	BLEU [%]	TER [%]
PBT normal	-	yes	34.8	50.7	33.3	51.9
PBT reversed		yes	34.9	50.6	33.0	51.9
HPBT normal		yes	35.8	50.5	34.1	51.7
HPBT reversed		yes	35.6	49.2	34.6	50.0
HPBT alignment-reversed	-	-	36.0	49.6	34.6	50.1
HPBT lm-reversed	-	-	35.6	50.4	34.0	51.6
HPBT alignment combination	iu	-	35.7	50.0	34.5	50.8
	intersection		35.7	50.2	34.6	51.3
	gdf		36.2	49.8	34.7	50.9
	union		36.2	49.4	34.8	50.7
HPBT phrase table combination	union	-	33.5	54.8	32.1	55.3
	intersection		35.1	50.1	34.0	51.1
	intersection +4 models		36.1	49.2	34.9	50.1
system combination setup #1	-	-	36.7	48.1	**35.0**	49.5

in TER compared to the best single system. All in all, alignment combination gives us no similar improvement like system combination. The combination of the phrase tables degrades the scores for the intersection heuristic as well as for the union heuristic. Adding the models of the reversed phrase table to the normal one yields an improvement of 0.3 points in BLEU. Compared to the system combination we lose 0.1 points in BLEU and 0.6 points in TER. Nevertheless, adding the four reversed model scores to the normal phrase table gives us the best result without system combination.

7.6.3 BOLT Chinese→English

Experimental results for the BOLT Chinese→English translation task are given in Table 7.8. The HPBT reversed system outperforms the HPBT normal system by 0.4 points in BLEU and 0.8 points in TER. The PBT reversed system performs worse with 0.4 points in BLEU and 0.1 points in TER compared to PBT normal. For both decoders, the source-reversed and target-reversed systems yield comparable results compared to the normal systems. PBT target-reversed performs worse with 0.8 points in BLEU and 2.4 points in TER compared to PBT normal. The alignment combination based on the iu heuristics is the best performing alignment combination, but is still behind the single systems. The phrase table combination yield performance difference of 0.2 points in BLEU and 0.5 points in TER compared to the HPBT reversed system. Nevertheless, system combination outperforms all systems by 0.6 points in BLEU and 0.3 points in TER compared to the best single engine HPBT reversed and even 1.0 points in BLEU and 0.9 points in TER compared to HPBT normal which is the best system trained only on the normal word order.

Table 7.8: Experimental results for BOLT Chinese→English translation task. The reversed setups yield small improvements compared to the normal setups. The system combination of all systems produce the best translation regarding to the automatic evaluation.

system	combination algorithm	syscomb setup #1	dev		test	
			BLEU [%]	TER [%]	BLEU [%]	TER [%]
HPBT normal	-	yes	17.2	64.0	16.0	63.6
HPBT reversed		yes	17.5	63.4	16.4	62.8
HPBT source-reversed		yes	17.5	63.9	16.0	63.6
HPBT target-reversed		yes	17.6	64.0	16.4	63.4
PBT normal	-	yes	17.2	64.5	16.2	63.8
PBT reversed		yes	17.1	64.1	15.8	63.9
PBT source-reversed		yes	17.2	64.5	15.9	64.5
PBT target-reversed		yes	16.6	66.3	15.4	66.2
HPBT alignment-reversed	-	-	17.5	63.5	16.4	62.9
HPBT lm-reversed			17.1	64.3	15.9	63.8
HPBT alignment combination	intersection	-	16.9	64.4	15.6	64.0
	union		17.5	64.1	16.4	63.3
	gdf		17.5	63.9	16.5	63.6
	iu		17.7	64.0	16.5	63.4
HPBT phrase table combination	intersection	-	17.5	63.5	16.0	63.1
	union		17.6	63.6	16.3	62.6
	union + 4 models		18.0	63.9	16.6	63.3
system combination setup #1	-	-	18.3	63.0	**17.0**	62.5

7.6.4 BOLT Arabic→English

Experimental results for the BOLT Arabic→English translation task are given in Table 7.9. We utilize both the phrase-based (PBT) and the hierarchical phrase-based (HPBT) engines. Both yield comparable results for the Arabic→English translation task. We achieve only small improvement of up to 0.2 BLEU points by applying either phrase table or alignment combinations. For Arabic→English, we utilize both decoders (PBT and HPBT) for the combination approaches, because both reach similar performance. Nevertheless, system combination is the choice of combination method also for Arabic→English. We improve translation quality by 0.3 points in BLEU by using the system combination implementation of Chapter 4.

7.7 Reversed Alignment vs. Reversed Language Model

In the previous chapters, we showed that a statistical machine translation system can benefit from reversing the word order. Furthermore, we showed that a system can also benefit by only training one component (e.g. language model, alignment model, or decoding direction) on a reversed corpora and integrate it into a normal order system. In this section, we will sum up from which component a

Table 7.9: Experimental results for BOLT Arabic→English.

system	combination algorithm	syscomb setup #1	dev		test	
			BLEU [%]	TER [%]	BLEU [%]	TER [%]
HPBT normal	-	yes	24.7	57.9	26.5	57.5
HPBT reversed		yes	24.9	57.5	26.6	57.3
HPBT source-reversed		yes	25.0	57.4	26.3	57.7
HPBT target-reversed		yes	24.9	58.1	26.6	57.7
PBT normal	-	yes	24.9	57.5	26.8	57.2
PBT reversed		yes	25.2	57.4	26.7	57.2
PBT source-reversed		yes	25.1	57.6	26.8	57.3
PBT target-reversed		yes	24.7	57.7	26.3	57.6
HPBT alignment combination	gdf	-	25.0	57.4	26.4	57.6
	intersection		24.9	57.6	26.7	57.5
	iu		24.8	57.3	26.5	57.6
	union		25.1	57.4	26.5	57.5
HPBT phrase table combination	union	-	25.0	57.3	26.4	57.5
	union +4 models		25.0	57.5	26.5	57.8
	intersection		25.0	57.5	26.6	57.6
	intersection +4 models		25.1	57.6	26.7	57.3
PBT alignment combination	gdf	-	24.9	57.5	26.6	57.4
	intersection		24.7	57.3	26.4	57.2
	iu		24.7	57.5	26.5	57.4
	union		24.9	58.1	26.8	57.4
PBT phrase table combination	union	-	24.9	57.5	26.7	57.4
	union +4 models		24.5	57.3	26.6	57.2
	intersection		24.6	57.6	26.4	57.6
	intersection +4 models		24.9	57.7	26.7	57.3
system combination setup #1	-	-	25.7	57.1	27.1	57.3

statistical machine translation engine can benefit by reversing the word order and which component only has a low affect by training on a different word order. We compare the normal, alignment-reversed, lm-reversed, and fully reversed system of both Chinese→English setups and the NTCIR-9 Japanese→English setup. The test set results are sum up in Table 7.10.

The difference between a normal system and a lm-reversed system is only the different language model. The alignment and phrase extraction are both utilized on the normal order corpora. Further, the HPBT decoding scheme is independent from the word order. If you compare the results of the normal setups with the lm-reversed setups, you can only see a small difference in the automatic scores. We come to the conclusion that the language model direction is not important for our statistical machine translation system.

The difference between a normal system and an alignment-reversed system is only the different

Table 7.10: HPBT normal, alignment-reversed, lm-reversed, and fully reversed systems of both Chinese→English setups and the NTCIR-9 Japanese→English setup.

	NTCIR-9 Japanese→English		**NTCIR-9 Chinese→English**		**BOLT Chinese→English**	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]	BLEU[%]	TER[%]
HPBT normal	30.7	63.9	34.1	51.7	16.0	63.6
HPBT lm-reverse	30.7	64.0	34.0	51.6	15.9	63.8
HPBT alignment-reverse	31.3	61.9	34.6	50.1	16.4	62.9
HPBT reverse	31.4	62.0	34.6	50.0	16.4	62.8

trained alignment. If we compare the results of the two different setups, we can see a difference in the automatic scores. Furthermore, we can see that the reversed trained alignment improves the translation quality of all three different setups. We come to the conclusion that the two language pairs Japanese→English and Chinese→English can benefit from a reversed trained alignment.

Finally, we compare the alignment-reversed systems with the fully reversed systems. Both systems are using the same alignment, but a different language model. Again by changing the forward language model into a backward language model, we can not see any change in the automatic scores of our translation engine.

7.8 Conclusion

In this work we revisited the idea of translation from right-to-left instead of the normal direction left-to-right. In order to do so, we did alignment and language model training as well as decoding for the reversed word order. We built up to eight systems with different translation directions and proposed to apply alignment combinations, phrase table combinations, and system combinations to take benefit of the strength of each of the setups. Without any changes in preprocessing and with the standard set of models, we achieved an improvement over normal phrase-based and hierarchical phrase-based setups. The improvement are 1.7 points in BLEU and 3.1 points in TER on the NTCIR-9 PatentMT task for Japanese→English. For NTCIR-9 Chinese→English we were 0.4 points in BLEU and 0.5 points in TER better than the best single system. For the BOLT Chinese→English data, we reached improvements of 1.0 points in BLEU and 0.9 points in TER.

In the analysis of our setups, we came to the conclusion that the trained alignment is the main reason for varying different translation results of systems built with different translation directions. The decoding algorithm has no effect for hierarchical phrase-based systems, but can lead to different results when using the pure phrase-based approach. The language model considers different context when trained on reversed corpora. The so called backward language model performance as good as the normal forward language model in our experiments.

We got only small improvements by the application of alignment or phrase table combination. Running system combination with eight normal, reversed and partial reversed systems yielded the highest improvements in translation quality. The system combinations are built on high-quality single engines and this approach has been successfully used in several evaluation setups of the RWTH Aachen University.

8 Evaluations

In this chapter, we will present the RWTH Aachen University system combination results in the most recent evaluation campaigns. All system combinations have been conducted as part of the EU-Bridge[1] project. EU-Bridge is a European research project which is aimed at developing innovative speech translation technology. Up to four research institutions involved in the EU-Bridge project combined their individual machine translation systems with the system combination approach presented in this thesis (Chapter 4) and participated with a joint setup in the machine translation track of the evaluation campaign at the *2014 International Workshop on Spoken Language Translation*[2] (IWSLT 2014) and in the shared translation task of the evaluation campaign at the *ACL 2014 Eighth Workshop on Statistical Machine Translation*[3] (WMT 2014). Additional to the automatic scores, we present the official HTER [Snover & Madnani$^+$ 09] human evaluation results for the IWSLT 2014 English→German and English→French text translation tasks.

8.1 Individual System Engines

In the following system combination setups, we will combine the individual systems from the RWTH Aachen University (RWTH), the University of Edinburgh (UEDIN), Karlsruhe Institute of Technology (KIT), and Fondazione Bruno Kessler (FBK). We will give a short introduction of the translation engines used by the four different research labs. Every research lab is maintaining their own translation engine whose outputs differ, but with comparable translation quality. For more details, we refer to the system papers of the joint submissions [Freitag & Wuebker$^+$ 14, Freitag & Peitz$^+$ 14].

[1] http://www.eu-bridge.eu
[2] http://workshop2014.iwslt.org/
[3] http://www.statmt.org/wmt14/

- RWTH Aachen University (RWTH):
  The state-of-the-art phrase-based baseline systems (RWTH PBT) and the hierarchical phrase-based systems (RWTH HPBT) are augmented with a hierarchical reordering model [Galley & Manning 08] and several additional language models. Additionally, the phrase-based systems use maximum expected BLEU training for phrasal, lexical and reordering models [He & Deng 12]. Further, RWTH Aachen University employs rescoring with recurrent neural language and translation models [Sundermeyer & Alkhouli$^+$ 14]. Both decoders are implemented in RWTH Aachen University's publicly available translation toolkit Jane [Vilar & Stein$^+$ 10, Wuebker & Huck$^+$ 12].

- University of Edinburgh (UEDIN):
  The University of Edinburgh translation engines are based on the open source translation toolkit Moses [Koehn & Hoang$^+$ 07a]. They set up phrase-based systems, and additionally string-to-tree syntax-based systems [Galley & Hopkins$^+$ 04]. Edinburgh applies factored models [Koehn & Hoang 07b]. Source side factors are words, POS tags, and Brown clusters (2000 classes). Target side factors are words, POS tags, Brown clusters (2000 classes) [Och 99], and morphological tags. For the syntax-based systems, the target-side data is parsed with different parsers, and right binarization is applied to the parse trees. Augmenting the system with non-syntactic phrases [Huck & Hoang$^+$ 14a] and adding soft source syntactic constraints [Huck & Hoang$^+$ 14b] yield further improvements.

- Karlsruhe Institute of Technology (KIT):
  Karlsruhe Institute of Technology translations are using an in-house phrase-based translations system [Vogel 03]. In all translation directions, they use a pre-reordering approach. Different reorderings of the source sentences are encoded in a word lattice. In addition, for the language pairs involving German, KIT applies different reorderings of both language pairs using a lexicalized reordering model [Koehn & Axelrod$^+$ 05]. In addition to the phrase table probabilities, Karlsruhe models the translation process by a bilingual language model [Niehues & Herrmann$^+$ 11] and a discriminative word lexicon using source context models [Niehues & Waibel 13].

- Fondazione Bruno Kessler (FBK):
  The Fondazione Bruno Kessler system is built upon a standard phrase-based system using the open source translation toolkit Moses [Koehn & Hoang$^+$ 07a]. Data selection is performed by the toolkit XenC [Rousseau 13] exploiting bilingual cross-entropy difference [Axelrod & He$^+$ 11] separately for each available training corpus. Different amount of texts are selected from each corpus ranging from 2% to 30%, and then concatenated for building one parallel corpus.

8.2 IWSLT 2014

In this section, we present the IWSLT 2014 [Cettolo & Niehues$^+$ 14] experimental results. The IWSLT is an annual scientific workshop associated with an open evaluation campaign on spoken language translation. For each task, monolingual and bilingual language resources are provided to participants in order to train their systems. The goal is to translate manual and automatic speech transcripts to combine the research fields of automatic speech recognition (ASR) and machine translation.

All reported BLEU [Papineni & Roukos$^+$ 02] and TER [Snover & Dorr$^+$ 06] scores are case-sensitive with one reference. In addition to the results on the development sets, we present the official evaluation results on *tst2014* for all translation tasks. These scores have been calculated by the organizers and can be found in the findings of the workshop. Furthermore, we compare our submissions

with other research group's submissions to the evaluation campaign. We only show results of submissions which yield comparable or better scores. All system combination results are generated with the implementation presented in this thesis (cf. Chapter 4).

8.2.1 IWSLT German→English SLT

The challenge of spoken language translation (SLT) is to translate automatic speech transcripts which have been previously recognized by an automatic speech recognition engine. Different to the text translation task, the source sentence can contain recognition errors which can affect the final translation quality. For the German→English SLT task, we combine three different individual systems generated by Karlsruhe Institute of Technology, University of Edinburgh, and RWTH Aachen University. Experimental results of the development set *dev2012* and the official evaluation set *tst2014* are given in Table 8.1. The system combination yields improvements of 0.8 points in BLEU and 0.1 points in TER compared to the best single system. All single systems as well as the system combination parameters are tuned on *dev2012*. For this year's IWSLT SLT track, *dev2012* is the only given test set containing automatic speech recognition output. No further official submission to the German→English SLT task has comparable translation quality compared to the systems in Table 8.1.

Table 8.1: Results for the IWSLT German→English SLT task. The system combination is a combination of 3 individual systems by Karlsruhe Institute of Technology, University of Edinburgh, and RWTH Aachen University.

system	**part of**	**dev2012**		**tst2014**	
	syscomb	BLEU[%]	TER[%]	BLEU[%]	TER[%]
Karlsruhe Institute of Technology	yes	20.7	60.5	18.3	63.9
RWTH Aachen University	yes	20.8	61.4	17.2	65.0
University of Edinburgh	yes	20.3	63.0	17.7	66.0
System combination	-	22.2	59.3	**19.1**	63.8

8.2.2 IWSLT German→English MT

The task of text translation is to translate written text that is grammatically and syntactically correct. Similar to the SLT track, the German→English MT system combination submission is a combined translation of three different individual systems by Karlsruhe Institute of Technology, University of Edinburgh, and RWTH Aachen University. Experimental results are given in Table 8.2. The system combination parameters are optimized on *tst2012*. The single engines are optimized on different combinations of *tst2010* and *tst2011*. Compared to the best individual system (RWTH Aachen University), the system combination improves translation scores by 0.8 points in BLEU and 0.9 points in TER on the official evaluation set *tst2014*. Seven different research labs submitted a run to the German→English MT task from which additional to the three single engines which are combined in our system combination setup, only one submission yields comparable translation quality. A system combination of three single engines by NTT Communication Science Laboratories (Kyoto, Japan) and Nara Institute of Science and Technology (Nara, Japan) (NAIST) yields better translation performance than the University of Edinburgh, but is still 2.0 points in BLEU and 1.8 points in TER behind our combined submission.

Table 8.2: Results for the IWSLT German→English MT task. The system combination is a combination of 3 systems by Karlsruhe Institute of Technology, University of Edinburgh, and RWTH Aachen University. The NTT-NAIST system is not part of the system combination.

system	part of syscomb	tst2010		tst2011		tst2012		tst2014	
		BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]
RWTH Aachen University	yes	31.8	47.2	38.3	41.3	32.0	47.0	25.0	55.5
Karlsruhe Institute of Technology	yes	31.5	47.6	37.1	42.5	32.0	47.6	24.6	55.6
NTT-NAIST	no	-	-	-	-	-	-	23.8	56.4
University of Edinburgh	yes	31.6	47.6	37.3	42.5	31.7	47.9	23.3	57.7
System combination	-	33.3	46.1	39.4	40.6	33.5	46.2	**25.8**	54.6

8.2.3 IWSLT English→French MT

For the English→French MT task, we combine five different individual systems. Karlsruhe Institute of Technology, Fondazione Bruno Kessler as well as RWTH Aachen University provide one individual system output for the system combination. The University of Edinburgh adds one contrastive system in addition to their primary system which is basically the same system trained only on a subset of the available training data. Experimental results are given in Table 8.3. The system combination of all five individual systems yields an improvement of up to 0.8 points in BLEU compared to the best individual system output on *tst2014*. Using a recurrent neural network (RNN) language model [Sundermeyer & Alkhouli$^+$ 14] to rescore a 1000-best list of the system combination output, leads to a small translation improvement of 0.1 points in BLEU on *tst2010*. The same RNN language model is applied in the individual system of RWTH Aachen University. The improvements are only small, as the model is already contained in one individual system. Nine systems were submitted to the IWSLT English→French translation task. The submission of MIT Lincoln Laboratory (Lexington, MA, USA) and Air Force Research Laboratory (Wright Patterson AFB, OH, USA) (MITLL-AFRL) yields comparable translation quality. The BLEU and TER scores can be seen in Table 8.3. The MITLL-AFRL system is 0.7 points in BLEU and 0.5 points in TER worse compared to the combined submission.

8.2.4 IWSLT English→German MT

For the English→German setup, we combine three different individual systems of the University of Edinburgh with the primary submission of Karlsruhe Institute of Technology. Additional to two phrase-based setups, the University of Edinburgh provides one string-to-tree syntactically augmented system. Experimental results are given in Table 8.4. All system combination parameters are tuned on *tst2012*. The single systems are tuned on different combinations of *tst2010* and *tst2011*. The final system combination submission enhances the translation quality by up to 0.6 points in BLEU and 0.4 points in TER compared to the best individual system on the official evaluation set *tst2014*. There is a total of 6 submission to the IWSLT English→German MT task from which only the NTT-NAIST system combination setup yields comparable translation quality. NTT-NAIST submission is a system combination of three different machine translation engines and yields worse results compared to all individual systems as well as to our combined submission.

Table 8.3: Results for the IWSLT English→French MT task. The system combination is a combination of 5 individual systems by Karlsruhe Institute of Technology, University of Edinburgh (2 systems), Fondazione Bruno Kessler, and RWTH Aachen University. The MITLL-AFRL system is not part of the system combination setup.

system	**part of syscomb**	**tst2010**		**tst2011**		**tst2012**		**tst2014**	
		BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]
Karlsruhe Institute of Technology	yes	33.1	48.4	37.3	42.5	39.1	40.2	36.2	45.2
University of Edinburgh primary	yes	33.6	48.5	40.2	40.6	41.0	39.6	35.9	45.8
University of Edinburgh secondary	yes	33.2	49.1	39.1	42.0	40.7	39.8	-	-
RWTH Aachen University	yes	34.5	47.6	41.1	40.1	42.0	38.6	35.7	44.5
MITLL-AFRL	no	-	-	-	-	-	-	35.5	45.7
Fondazione Bruno Kessler	yes	32.8	50.4	39.2	42.6	40.0	41.4	34.2	46.8
System combination	-	35.1	48.5	41.7	41.4	44.0	38.7	**37.0**	45.2
+RNN LM *n*-best rescoring	-	35.2	48.5	41.7	41.3	44.3	38.5	-	-

Table 8.4: Results for the IWSLT English→German MT task. The system combination is a combination of 4 individual systems by Karlsruhe Institute of Technology and University of Edinburgh (3 systems). The NTT-NAIST system is not part of the combination.

system	**part of syscomb**	**tst2010**		**tst2011**		**tst2012**		**tst2014**	
		BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]
Karlsruhe Institute of Technology	yes	24.5	55.2	27.1	50.5	23.5	56.0	22.7	57.7
University of Edinburgh primary	yes	24.9	55.5	27.8	50.1	23.4	56.9	22.6	59.0
University of Edinburgh secondary	yes	24.1	55.7	26.7	50.8	22.2	57.3	-	-
University of Edinburgh syntax	yes	24.8	55.3	26.5	50.5	23.1	56.6	-	-
NTT-NAIST	no	-	-	-	-	-	-	22.1	57.6
System combination -		25.9	54.0	28.1	49.1	24.9	55.0	**23.3**	57.3

8.2.5 IWSLT Human Evaluation Results

As part of the IWSLT 2014 evaluation campaign, human interacted scoring is performed for the English→German MT as well as for the English→French MT task. Human evaluation is based on post-editing and HTER (Human-mediated Translation Edit Rate) [Snover & Madnani$^+$ 09]. The manual correction of the machine translation output is called post-editing. The HTER score consists of measuring the minimum edit distance between the machine translation and its manually post-edited version. For further details, we refer to the paper of the IWSLT 2014 evaluation campaign. Human evaluation is performed only on a subset of *tst2013*, namely on 628 segments for English→German and 622 segments for English→French, both corresponding to around 11,000 words.

IWSLT English→German MT: The HTER results for the English→German MT task are given in Table 8.5. In this task, the HTER scores give us the same system ranks as given by the BLEU scores in the automatic evaluation (cf. Section 8.2.4). The system combination generated from the three different systems of University of Edinburgh and the primary system of Karlsruhe Institute of Technology yields improvement of 0.7 points in HTER compared to the best single system.

Table 8.5: Human evaluation results for the IWSLT English→German MT task. The system combination is a combination of 4 individual systems by Karlsruhe Institute of Technology and University of Edinburgh (3 systems). The NTT-NAIST system is not part of the system combination setup.

system	HTER [%]
Karlsruhe Institute of Technology	19.9
University of Edinburgh	20.9
NTT-NAIST	21.3
System combination	19.2

IWSLT English→French MT: The HTER results for the English→French MT task are given in Table 8.6. Different to the HTER results of the previous section, the human evaluated scores lead to different system ranks compared to the automatic evaluation of Section 8.2.3. Regarding to the human judgement, RWTH Aachen University system performs 1.6 points better in HTER compared to the system of Karlsruhe Institute of Technology, which yields better performance regarding the automatically calculated BLEU scores. Nevertheless, the system combination setup performs better than all other systems regarding HTER as well as regarding the automatic calculated BLEU scores.

Table 8.6: Human evaluation results for the IWSLT English→French MT task. The system combination is a combination of 5 individual systems by Karlsruhe Institute of Technology, University of Edinburgh (2 systems), Fondazione Bruno Kessler, and RWTH Aachen University. The MITLL-AFRL system is not part of the system combination setup.

system	HTER [%]
RWTH Aachen University	19.3
Karlsruhe Institute of Technology	20.9
University of Edinburgh	21.5
MITLL-AFRL	22.6
Fondazione Bruno Kessler	22.9
System combination	19.2

8.3 WMT 2014

In this section, we present the official results of the shared translation task of the evaluation campaign at the WMT 2014 [Bojar & Buck$^+$ 14]. As part of the evaluation campaign, the organizers

provide parallel corpora to train translation models, monolingual corpora to train language models, and development sets to tune system parameters. The test data for this task is selected from news stories from online sources. We participate in two translation tasks, namely German→English and English→German. The weights of the individual system engines are optimized on different test sets which partially or fully include *newstest2011* or *newstest2012*. System combination weights are either optimized on newstest2011 or newstest2012. We keep *newstest2013* as an unseen test set which is not used for tuning the system combination or any of the individual systems. The given results for the *newstest2014* test set are the official evaluation results provided by the organizers. All reported BLEU [Papineni & Roukos$^+$ 02] and TER [Snover & Dorr$^+$ 06] scores are case-sensitive, calculated with one reference. All combined systems are generated by the system combination implementation presented in this thesis (cf. Chapter 4).

8.3.1 WMT German→English

The automatic scores of all individual systems as well as of the final system combination submission are given in Table 8.7. Karlsruhe Institute of Technology, University of Edinburgh, and RWTH Aachen University are each providing one individual phrase-based system output. RWTH Aachen University (*HPBT*) and University of Edinburgh (*syntax*) are providing additional systems based on either the hierarchical translation approach or a string-to-tree syntax model. For German→English, the system combination parameters are optimized on *newstest2012*. System combination gives us a gain of 0.8 points in BLEU and 2.7 points in TER for *newstest2014* compared to the best single system. There were three additional evaluation submissions that yield comparable scores and were not part of our system combination setup. First, a joint submission of LIMSI-CNRS (Orsay, France) and Karlsruhe Institute of Technology (Karlsruhe, Germany). Second, a single engine of Carnegie Mellon University (Pittsburgh, PA, USA) . Third, a system combination submission of several systems from Dublin City University (Dublin, Ireland) and Institute of Computing Technology Chinese Academy of Sciences (Beijing, China) (DCU-ICTCAS). All three submission are of less translation quality regarding BLEU and TER compared to our system combination setup as well as to the best single engine of University of Edinburgh.

8.3.2 WMT English→German

The results of all English→German system setups are given in Table 8.8. Karlsruhe Institute of Technology is providing a phrase-based system output, the University of Edinburgh is providing two phrase-based system outputs and six syntax-based ones to the system combination setup. *University of Edinburgh primary* is a phrase-based setup. *University of Edinburgh secondary* is a phrase-based setup using fewer models. The different syntax-based systems do not only differ in the applied syntax parser, they also differ by either the source side, the target side, or both sides being parsed. For English→German, the system combination parameters are optimized on *newstest2011*. Combining all nine different system outputs yields an improvement of 0.5 points in BLEU over the best single system performance. From this result, we can conclude that system combination is a reliable method to improve the translation quality even when the individual input systems differ only slightly. There are two additional evaluation systems, which yield compareable performance: both Stanford University (Stanford, CA, USA) and Uppsala University (Uppsala, Sweden) submitted one singe engine to the evaluation campaign. Nevertheless, both system performances are worse in comparison to our best single engine and system combination submission.

Table 8.7: Results for the WMT German→English translation task. The system combination is tuned on newstest2012, newstest2013 is used as held-out test set for all individual systems and system combination. All cursive written systems are not generated by our partners and are not part of the system combination.

system	**part of syscomb**	**newstest 2011**		**newstest 2012**		**newstest 2013**		**newstest 2014**	
		BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]
University of Edinburgh syntax	yes	23.0	60.1	23.2	60.8	26.2	56.9	28.2	60.6
University of Edinburgh primary	yes	23.9	59.2	24.7	58.3	27.4	55.0	28.0	59.9
LIMSI-KIT	no	-	-	-	-	-	-	27.5	59.4
Carnegie Mellon University	no	-	-	-	-	-	-	27.1	59.2
RWTH Aachen University PBT	yes	23.6	59.5	24.2	58.5	27.0	55.0	27.0	60.3
Karlsruhe Institute of Technology	yes	25.0	57.6	25.2	57.4	27.5	54.4	26.9	59.7
RWTH Aachen University HPBT	yes	23.3	59.9	24.1	59.0	26.7	55.9	26.8	60.9
DCU-ICTCAS	no	-	-	-	-	-	-	26.5	60.8
System combination	-	25.6	57.1	26.4	56.5	29.1	53.4	**29.0**	57.9

8.4 Conclusion

In this chapter, we presented the official evaluation results on the recent machine translation tracks of the evaluation campaign at the *2014 International Workshop on Spoken Language Translation* and in the shared translation task of the evaluation campaign at the *ACL 2014 Eighth Workshop on Statistical Machine Translation*. In addition to the automatic calculated BLEU and TER scores, we presented human evaluation results based on the HTER metric. Translation quality has been improved by the application of system combination on all participated tasks. Moreover, the system combination submission was the winner throughout all text translation and spoken language translation tasks. Furthermore, we also yielded improvements regarding the human evaluated HTER scores by the application of system combination. Summarizing the results from this chapter, we implemented a reliable toolkit for combining different machine translation hypotheses which outperforms all submissions in the recent evaluation campaigns.

Table 8.8: Results for the WMT English→German translation task. The system combination is tuned on *newstest2011*, *newstest2013* is used as held-out test set for all individual systems and system combination. The system combination is generated by 9 individual systems. All cursive written systems are not part of the system combination setup.

system	**part of syscomb**	**newstest 2011**		**newstest 2012**		**newstest 2013**		**newstest 2014**	
		BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]
University of Edinburgh primary	yes	17.8	66.9	18.5	64.6	20.8	62.3	20.1	70.8
University of Edinburgh secondary	yes	17.5	67.3	18.2	65.0	20.5	62.7	-	-
University of Edinburgh S2T (ParZu)	yes	17.2	67.6	18.0	65.5	20.2	62.8	20.1	69.9
Stanford University	no	-	-	-	-	-	-	20.0	71.0
Karlsruhe Institute of Technology	yes	17.1	67.0	17.8	64.8	20.2	62.2	19.5	70.8
Uppsala University	no	-	-	-	-	-	-	19.0	72.8
University of Edinburgh T2S (Berkeley)	yes	16.7	68.9	17.5	66.9	19.5	63.8	18.8	73.3
University of Edinburgh S2T (BitPar)	yes	16.3	69.0	17.3	66.6	19.5	63.9	18.6	73.4
University of Edinburgh S2T (Berkeley)	yes	16.3	68.9	17.2	66.7	19.3	63.8	18.6	73.2
University of Edinburgh S2T (Stanford)	yes	16.1	69.2	17.2	67.0	19.0	64.2	18.3	73.7
University of Edinburgh S2S (Berkeley)	yes	16.3	69.2	17.3	66.8	19.1	64.3	18.2	74.3
System combination	-	18.4	65.0	18.7	63.4	21.3	60.6	**20.6**	70.5

9 Conclusion and Scientific Achievements

In this chapter we revisit the scientific goals that we defined in Chapter 2 and analyze in how far we accomplished them:

- In Chapter 4, we invented a novel combination aproach which is integrated into RWTH's open source statistical machine translation toolkit Jane. We found that our novel system combination aproach performs on a similar level or better than the best evaluation system combination submissions on all WMT 2011 system combination shared task language pairs (with English as target language). We got the highest improvement of 0.7 points in BLEU for Spanish→English when adding both the big LM and IBM-1 features. Adding the big LM over the baseline enhanced the translation quality for all four language pairs. Adding IBM-1 lexicon models on top of the big LM was of marginal or no benefit for most language pairs, but at least provided slight improvements for Spanish→English. Furthermore, we introduced governed insertion and showed improvements of the translation quality for Arabic→English, Spanish→English, and German→English while keeping translation quality for the other language pairs.

- We introduced a novel local system voting model trained by a feedforward neural network in Chapter 5. In contrast to the traditional system voting features, the presented local system voting model takes the word contents and their combinatorial occurrences into account and does not only promote global preferences for some individual systems. This advantage gives confusion network decoding the option to prefer other systems at different positions even in the same sentence. As all words are projected to a continuous space, the neural network gives also unseen word sequences a useful probability. Due to the relatively small neural network training set, we used word classes in some experiments to tackle the data sparsity problem. Training an additional local voting model by a neural network with word classes yields translation improvement of 0.9 points in BLEU and 0.5 points in TER for the BOLT Chinese→English and Arabic→English translation tasks. We analyzed the translation results and the functionality of the novel model. The occurrence distribution showed that words which have been produced by only few input systems were more likely to be part of the syscom output when using the proposed model.

- In Chapter 6, we introduced a novel alignment approach which aligns the individual system engines into a lattice from which the consensus translation can be extracted. We used the phrase

information of all individual systems to align the different machine translation systems. By doing so, we solved many alignment errors which occur in a target-based system combination as wrong word repetitions, lack of *m*-to-*n* alignments, empty translations, multiple translations of the same source sequence or simple word-to-word alignment errors. Experiments showed translation quality improvement of 0.6 points in BLEU and 0.5 points in TER for the BOLT Chinese→English and Arabic→English translation tasks. The final translation output of the proposed lattice decoding approach did not only improve the automatic metrics, but also augments the fluency of the output. We defined error classes for the different lattice types and conducted human evaluation of the lattices. We came to the conclusion that the novel lattices produced less errors compared to the traditional confusion networks.

- We investigated the decoding directions for the two commonly known statistical machine translation approaches phrase-based translation and hierarchical phrase-based translation in Chapter 7. We reversed the word order of the source and/or target language and compared the translation results with the normal direction. We did alignment and language model training as well as decoding for the reverse and partial reverse word orders. We built up several systems with different translation directions and proposed to apply alignment combinations, phrase table combinations, and system combinations to take benefit of the strength of each of the setups. Without any changes in preprocessing and with the standard set of models, we achieved improvements over pure phrase-based and hierarchical phrase-based setups. We achieved gains of 1.7 points in BLEU and 3.1 points in TER on the NTCIR-9 PatentMT task for Japanese→English. For NTCIR-9 Chinese→English we were 0.4 points in BLEU and 0.5 points in TER better than the best single system. For the BOLT Chinese→English data, we reached improvements of 1.0 points in BLEU and 0.9 points in TER. We got only small improvements by combining the reverse and normal alignments. The phrase table combinations of the reverse and normal trained phrase tables degrades the translation quality. Nevertheless, if we add the model scores of the reverse phrase table to the normal one, we get the best results without system combination. Running system combination with several normal, reverse and partial reverse systems yielded the highest improvements in translation quality.

- We presented the recent evaluation results which were obtained with the system combination approach invented in this thesis in Chapter 8. We compared our translation setups with the translation engines of world-leading research labs all over the world and reached first position in various language pairs.

9.1 Concluding Remarks

In general terms, we investigated various methods for combining the benefits of different machine translation engines. We started with inventing a novel combination approach which outperformed the system combination approaches of all other research labs. We introduced several new extensions which enhanced the translation quality on top of the previously invented approach. In addition to the automatic scores, we also conducted human evaluation to show the benefits of the proposed methods. We investigated the word order directions of both the phrase-based and hierarchical phrase-based machine translation approaches. We showed how to yield improvements with system combination conducted on systems based on different decoding directions. Finally, we presented the recent evaluation results obtained with the proposed approaches of this thesis.

As an additional result of the work carried out for this thesis, the open source machine translation toolkit `Jane` has been extended. The toolkit contains the implementation of all the methods described in this thesis and can be used for reproducing the results presented. The availability of the code also

allows other researches to build upon the material presented on this thesis, by which we hope to provide a contribution to the scientific community.

10 Corpus Statistics

The corpus statistics of the available training data for the different translations tasks are presented in this chapter. Each individual system has the choice to use all or only a subset of the available training data to build its system. The data is given by either the organizers of the evaluation campaign (WMT/ IWSLT/ NTCIR-9) or the project (BOLT) itself. All systems in this thesis can be categories as 'constraint' system, which means that no additional external data has been used for building the systems. We refer to the findings of the evaluation campaigns for more corpus details: WMT 2011 [Callison-Burch & Koehn$^+$ 11], IWSLT 2014 [Cettolo & Niehues$^+$ 14], NTCIR-9 [Goto & Lu$^+$ 11].

WMT French→English

The corpus statistics for the WMT French→English translation task can be found in Table 10.1. The numbers are calculated on the complete set of all available bilingual training data provided by the organizers. In general the training data comes from 4 different sources: Europarl (1.8M), News Commentary (115k), United Nations (12.3M), and 10^9 (22.5M). All system combinations are tuned on *syscomtune* and tested on *syscomtest*.

Table 10.1: Corpus statistics WMT French→English.

	French	English
Sentences	37M	
Running Words	1262M	1066M
Vocabulary	2.8M	2.9M
Syscomtune sentences	1003	
Syscomtest sentences	2000	

WMT German→English

The corpus statistics for the WMT German→English translation task are listed in Table 10.2. The bilingual training data is generated from two different sources, namely Europarl (1.7M) and News Commentary (136k). The system combination parameters are tuned on *syscomtune* and tested on *syscomtest*.

Table 10.2: Corpus statistics WMT German→English.

	German	English
Sentences	1.9M	
Running Words	49M	51M
Vocabulary	390K	120K
Syscomtune sentences	1003	
Syscomtest sentences	2000	

WMT Spanish→English

The corpus statistics for the WMT Spanish→English translation task can be found in Table 10.3. The sources for creating the total 12 million training data are News Commentary (130k), Europarl (1.8M), and United Nations (10.6M). *syscomtune* is used for optimizing the system combination parameters and *syscomtest* is used to evaluate the results.

Table 10.3: Corpus Statistics WMT Spanish→English.

	Spanish	English
Sentences	12.5M	
Running Words	402M	357M
Vocabulary	370K	320K
Syscomtune sentences	1003	
Syscomtest sentences	2000	

WMT Czech→English

The corpus statistics for the WMT Czech→English are shown in Table 10.4. The sources from which the entire data comes from are: News Commentary (122K), Europarl (460K), and CzEng (7.2M). All system combinations are optimized on *syscomtune*.

BOLT Chinese→English

The corpus for Chinese→English translations is taken from the BOLT project and consists of text drawn from ”discussion forums” in Mandarin Chinese. Table 10.5 shows the statistics of the data.

Table 10.4: Corpus statistics WMT Czech→English.

	Czech	English
Sentences	7.8M	
Running Words	86M	100M
Vocabulary	75M	87M
Syscomtune sentences	1003	
Syscomtest sentences	2000	

Table 10.5: Corpus statistics BOLT Chinese→English.

	Chinese	English
Sentences	13M	
Running Words	255M	279M
Vocabulary	370K	833K
Tune sentences (NN)	1844	
Tune sentences (MERT)	985	
Test sentences	1124	

BOLT Arabic→English

For Arabic→English, we used the current BOLT data set (corpus statistics are given in Table 10.6). The test sets consist of text drawn from ”discussion forums” in Egyptian Arabic.

Table 10.6: Corpus statistics BOLT Arabic→English.

	Arabic	English
Sentences	8M	
Running Words	189M	186M
Vocabulary	608K	519K
Tune sentences (NN)	1510	
Tune sentences (MERT)	1080	
Test sentences	1137	

NTCIR-9 Japanese→English

For Japanese→English, all our experiments were conducted on the NTCIR-9 [1] PatentMT data. NTCIR-9 is a machine translation evaluation task for patent domain. We used the evaluation data provided by the organizers (corpus statistics are given in Table 10.7). The training data was built from patent documents published between 1993 and 2005. The development data consists of a total of 2000 sentence pairs built from patent documents published in 2006 and 2007.

Table 10.7: Corpus statistics NTCIR-9 Japanese→English.

	Japanese	English
Sentences	3.2M	
Running Words	109M	109M
Vocabulary	122K	112K
Tune sentences	1000	
Test sentences	1000	

NTCIR-9 Chinese→English

For Chinese→English, we used the evaluation data provided by the organizers of the NTCIR-9 conference (corpus statistics are given in Table 10.8). The data was mostly from patent description sentences (Patent documents consist of a title, abstract, claim, and description.) Those sentence pairs from patents published on or prior to 2005 were used for the training data, while those on or after 2006 were used for the development and test data.

Table 10.8: Corpus statistics NTCIR-9 Chinese→English.

	Chinese	English
Sentences	1M	
Running Words	41M	43M
Vocabulary	95K	315K
Tune sentences	1000	
Test sentences	1000	

[1] http://research.nii.ac.jp/ntcir/ntcir-9/index.html

List of Figures

List of Tables

Curriculum Vitæ

Personal Details

Name	Markus Freitag
Email	freitagmarkus@gmx.de
Date of birth	January 04, 1986
Place of birth	Radolfzell, Germany
Nationality	German

Education and Work Experience

Jul 2005 – Sep 2005	Internship at Gerling UK, London
Oct 2005 – Jun 2010	Study of Computer Science at RWTH Aachen University Degree: Diplom-Informatiker
Apr 2013 – Jun 2013	Internship at IBM Research, Yorktown (New York)
Jun 2010 – Jul 2015	Research Assistant at RWTH Aachen University, Human Language Technology and Pattern Recognition Group
Aug 2015 – present	Researcher at IBM Research, Yorktown (New York)

Bibliography

[Allauzen & Riley+ 07] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, M. Mohri: OpenFst: A General and Efficient Weighted Finite-State Transducer Library. In J. Holub, J. Zdárek, editors, *Implementation and Application of Automata*, Vol. 4783 of *Lecture Notes in Computer Science*, pp. 11–23. Springer Berlin Heidelberg, 2007.

[Axelrod & He+ 11] A. Axelrod, X. He, J. Gao: Domain Adaptation via Pseudo In-Domain Data Selection. *Conference on Empirical Methods in Natural Language (EMNLP)*, pp. 355–362, Edinburgh, UK, July 2011.

[Bangalore & Bordel+ 01] S. Bangalore, G. Bordel, G. Riccardi: Computing Consensus Translation from Multiple Machine Translation Systems. *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 351–354, Madonna di Campiglio, Italy, December 2001.

[Bar-Hillel 51] Y. Bar-Hillel: The Present State of Research on Mechanical Translation. *American Documentation*, Vol. 2, pp. 229–237, 1951.

[Barrault 10] L. Barrault: Many: Open Source Machine Translation System Combination. *The Prague Bulletin of Mathematical Linguistics*, Vol. 93, pp. 147–155, January 2010.

[Bengio & Schwenk+ 06] Y. Bengio, H. Schwenk, J.S. Senécal, F. Morin, J.L. Gauvain: Neural probabilistic language models. In *Innovations in Machine Learning*, pp. 137–186. Springer, 2006.

[Bojar & Buck+ 14] O. Bojar, C. Buck, C. Federmann, B. Haddow, P. Koehn, J. Leveling, C. Monz, P. Pecina, M. Post, H. Saint-Amand, R. Soricut, L. Specia, A. Tamchyna: Workshop on Statistical Machine Translation (WMT). *Ninth Workshop on Statistical Machine Translation*, pp. 12–58, Baltimore, MD, USA, June 2014. Association for Computational Linguistics.

[Boudahmane & Buschbeck+ 11] K. Boudahmane, B. Buschbeck, E. Cho, J.M. Crego, M. Freitag, T. Lavergne, H. Ney, J. Niehues, S. Peitz, J. Senellart, A. Sokolov, A. Waibel, T. Wandmacher, J. Wuebker, F. Yvon: Advances on Spoken Language Translation in the Quaero Program. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 114–120, San Francisco, California, USA, December 2011.

[Brown & Della Pietra+ 93] P.F. Brown, S.A. Della Pietra, V.J. Della Pietra, R.L. Mercer: The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, Vol. 19, No. 2, pp. 263–311, June 1993.

[Callison-Burch & Koehn+ 11] C. Callison-Burch, P. Koehn, C. Monz, O.F. Zaidan: Findings of the 2011 Workshop on Statistical Machine Translation. *Sixth Workshop on Statistical Machine Translation (WMT)*, pp. 22–64, Edingburgh, UK, July 2011.

[Cettolo & Niehues+ 14] M. Cettolo, J. Niehues, S. Stücker, L. Bentivogli, M. Federico: Report on the 11th IWSLT Evaluation Campaign, IWSLT 2014. M. Federico, S. Stücker, F. Yvon, editors,

Eleventh International Workshop on Spoken Language Translation (IWSLT), pp. 2–17, Lake Tahoe, CA, USA, December 2014.

[Chen & Kuhn$^+$ 11] B. Chen, R. Kuhn, G. Foster, H. Johnson: Unpacking and transforming feature functions: New ways to smooth phrase tables. *Machine Translation Summit (MT Summit XIII)*, pp. 269–275, Xiamen, China, September 2011.

[Chiang 05] D. Chiang: A hierarchical Phrase-based Model for Statistical Machine Translation. *43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 263–270, Ann Arbor, MI, USA, June 2005.

[Chiang 07] D. Chiang: Hierarchical Phrase-Based Translation. *Computational Linguistics*, Vol. 33, No. 2, pp. 201–228, 2007.

[Denkowski & Lavie 11] M. Denkowski, A. Lavie: Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. *Sixh Workshop on Statistical Machine Translation (WMT)*, pp. 85–91, Edingburgh, UK, July 2011.

[Denkowski & Lavie 14] M. Denkowski, A. Lavie: Meteor Universal: Language Specific Translation Evaluation for Any Target Language. *Ninth Workshop on Statistical Machine Translation (WMT). Association for Computational Linguistics*, pp. 376–380, Baltimore, MD, USA, June 2014.

[Du & Ma$^+$ 09] J. Du, Y. Ma, A. Way: Source-side context-informed hypothesis alignment for combining outputs from machine translation systems. *Twelfth Machine Translation Summit (MT Summit XII)*, pp. 230–237, Ottawa, ON, Canada, August 2009.

[Duchateau & Demuynck$^+$ 02] J. Duchateau, K. Demuynck, P. Wambacq: Confidence scoring based on backward language models. *Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 1, pp. I–221–I–224, Orlando, FL, USA, May 2002.

[Feng & Freitag$^+$ 13] M. Feng, M. Freitag, H. Ney, B. Buschbeck, J. Senellart, J. Yang: The System Combination RWTH Aachen: SYSTRAN for the NTCIR-10 PatentMT Evaluation. *10th NTCIR Conference*, pp. 301–308, Tokyo, Japan, June 2013.

[Feng & Liu$^+$ 09] Y. Feng, Y. Liu, H. Mi, Q. Liu, Y. Lü: Lattice-based system combination for statistical machine translation. *2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1105–1113. Association for Computational Linguistics, August 2009.

[Feng & Schmidt$^+$ 11] M. Feng, C. Schmidt, J. Wuebker, S. Peitz, M. Freitag, H. Ney: The RWTH Aachen System for NTCIR-9 PatentMT. *Ninth NTCIR Workshop Meeting*, pp. 600–605, Tokyo, Japan, December 2011.

[Feng & Schmidt$^+$ 13] M. Feng, C. Schmidt, J. Wuebker, M. Freitag, H. Ney: The RWTH Aachen System for NTCIR-10 PatentMT. *10th NTCIR Conference*, pp. 309–314, Tokyo, Japan, June 2013.

[Finch & Sumita 09] A. Finch, E. Sumita: Bidirectional phrase-based statistical machine translation. *2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1124–1132, Singapore, August 2009. Association for Computational Linguistics.

[Fiscus 97] J. Fiscus: A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER). *1997 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 347–354, December 1997.

[Frederking & Nirenburg 94] R. Frederking, S. Nirenburg: Three heads are better than one. *Fourth Conference on Applied Natural Language Processing (ANLP)*, pp. 95–100, Stuttgart, Germany, October 1994. Association for Computational Linguistics.

[Freitag & Feng+ 13] M. Freitag, M. Feng, M. Huck, S. Peitz, H. Ney: Reverse Word Order Models. *Machine Translation Summit*, pp. 159–166, Nice, France, September 2013.

[Freitag & Huck+ 14] M. Freitag, M. Huck, H. Ney: Jane: Open Source Machine Translation System Combination. *Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 29–32, Gothenburg, Sweden, April 2014. Association for Computational Linguistics.

[Freitag & Leusch+ 11] M. Freitag, G. Leusch, J. Wuebker, S. Peitz, H. Ney, T. Herrmann, J. Niehues, A. Waibel, A. Allauzen, G. Adda, J.M. Crego, B. Buschbeck, T. Wandmacher, J. Senellart: Joint WMT Submission of the QUAERO Project. *Sixth Workshop on Statistical Machine Translation (WMT)*, pp. 358–364, Edinburgh, UK, July 2011.

[Freitag & Peitz+ 12] M. Freitag, S. Peitz, M. Huck, H. Ney, T. Herrmann, J. Niehues, A. Waibel, A. Allauzen, G. Adda, B. Buschbeck, J.M. Crego, J. Senellart: Joint WMT 2012 Submission of the QUAERO Project. *NAACL 2012 Seventh Workshop on Statistical Machine Translation (WMT)*, pp. 322–329, Montreal, Canada, June 2012.

[Freitag & Peitz+ 13] M. Freitag, S. Peitz, J. Wuebker, H. Ney, N. Durrani, M. Huck, P. Koehn, T.L. Ha, J. Niehues, M. Mediani, T. Herrmann, A. Waibel, N. Bertoldi, M. Cettolo, M. Federico: EU-BRIDGE MT: Text Translation of Talks in the EU-BRIDGE Project. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 128–135, Heidelberg, Germany, December 2013.

[Freitag & Peitz+ 14] M. Freitag, S. Peitz, J. Wuebker, H. Ney, M. Huck, R. Sennrich, N. Durrani, M. Nadejde, P. Williams, P. Koehn, T. Herrmann, E. Cho, A. Waibel: EU-BRIDGE MT: Combined Machine Translation. *ACL 2014 Ninth Workshop on Statistical Machine Translation (WMT)*, pp. 105–113, Baltimore, Maryland, USA, June 2014.

[Freitag & Peter+ 15] M. Freitag, J.T. Peter, S. Peitz, M. Feng, H. Ney: Local System Voting Feature for Machine Translation System Combination. *EMNLP 2015 Tenth Workshop on Statistical Machine Translation*, pp. 467—476, Lisbon, Portugal, September 2015.

[Freitag & Wuebker+ 14] M. Freitag, J. Wuebker, S. Peitz, H. Ney, M. Huck, A. Birch, N. Durrani, P. Koehn, M. Mediani, I. Slawik, J. Niehues, E. Cho, A. Waibel, N. Bertoldi, M. Cettolo, M. Federico: Combined Spoken Language Translation. *Eleventh International Workshop on Spoken Language Translation (IWSLT)*, pp. 57–64, Lake Tahoe, CA, USA, December 2014.

[Frinken & Fornés+ 12] V. Frinken, A. Fornés, J. Lladós, J.M. Ogier: Bidirectional Language Model for Handwriting Recognition. *Proceedings of the 2012 Joint IAPR International Conference on Structural, Syntactic, and Statistical Pattern Recognition*, pp. 611–619, Hiroshima, Japan, November 2012. Springer-Verlag.

[Galley & Hopkins+ 04] M. Galley, M. Hopkins, K. Knight, D. Marcu: What's in a translation rule? *Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL)*, pp. 273–280, Boston, MA, USA, May 2004.

[Galley & Manning 08] M. Galley, C.D. Manning: A simple and effective hierarchical phrase reordering model. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 848–856, Honolulu, HI, USA, October 2008. Association for Computational Linguistics.

[Goto & Lu$^+$ 11] I. Goto, B. Lu, K.P. Chow, E. Sumita, B.K. Tsou: Overview of the Patent Machine Translation Task at the NTCIR-9 Workshop. *The 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, pp. 559–578, Tokyo, Japan, December 2011.

[He & Deng 12] X. He, L. Deng: Maximum expected bleu training of phrase and lexicon translation models. *50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 292–301, Jesu Island, South Korea, July 2012. Association for Computational Linguistics.

[He & Toutanova 09] X. He, K. Toutanova: Joint optimization for machine translation system combination. *Conference on Empirical Methods in Natural Language Processing (EMNLP): Volume 3*, pp. 1202–1211, Singapore, August 2009.

[He & Yang$^+$ 08] X. He, M. Yang, J. Gao, P. Nguyen, R. Moore: Indirect-HMM-based Hypothesis Alignment for Combining Outputs from Machine Translation Systems. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 98–107, Honolulu, HI, USA, October 2008.

[Heafield & Lavie 10] K. Heafield, A. Lavie: Combining Machine Translation Output with Open Source: The Carnegie Mellon Multi-Engine Machine Translation Scheme. *The Prague Bulletin of Mathematical Linguistics*, Vol. 93, pp. 27–36, January 2010.

[Hildebrand & Vogel 08] A.S. Hildebrand, S. Vogel: Combination of Machine Translation Systems via Hypothesis Selection from Combined N-Best Lists. *Eighth Conference of the Association for Machine Translation in the Americas (AMTA)*, pp. 254–261, Waikiki, HI, USA, October 2008.

[Hillard & Hoffmeister$^+$ 07] D. Hillard, B. Hoffmeister, M. Ostendorf, R. Schlüter, H. Ney: i ROVER: improving system combination with classification. *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pp. 65–68, Rochester, NY, USA, April 2007. Association for Computational Linguistics.

[Huck & Hoang$^+$ 14a] M. Huck, H. Hoang, P. Koehn: Augmenting String-to-Tree and Tree-to-String Translation with Non-Syntactic Phrases. *Ninth Workshop on Statistical Machine Translation (WMT)*, pp. 486–498, Baltimore, MD, USA, June 2014.

[Huck & Hoang$^+$ 14b] M. Huck, H. Hoang, P. Koehn: Preference Grammars and Soft Syntactic Constraints for GHKM Syntax-based Statistical Machine Translation. *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)*, pp. 148–156, Doha, Qatar, October 2014.

[Huck & Peitz$^+$ 12a] M. Huck, S. Peitz, M. Freitag, H. Ney: Discriminative Reordering Extensions for Hierarchical Phrase-Based Machine Translation. *16th Annual Conference of the European Association for Machine Translation (EAMT)*, pp. 313–320, Trento, Italy, May 2012.

[Huck & Peitz$^+$ 12b] M. Huck, S. Peitz, M. Freitag, M. Nuhn, H. Ney: The RWTH Aachen Machine Translation System for WMT 2012. *NAACL 2012 Seventh Workshop on Statistical Machine Translation*, pp. 304–311, Montreal, Canada, June 2012.

[Huck & Peter$^+$ 12] M. Huck, J.T. Peter, M. Freitag, S. Peitz, H. Ney: Hierarchical Phrase-Based Translation with Jane 2. *The Prague Bulletin of Mathematical Linguistics*, Vol. 98, pp. 37–50, October 2012.

[Huck & Vilar+ 13] M. Huck, D. Vilar, M. Freitag, H. Ney: A Performance Study of Cube Pruning for Large-Scale Hierarchical Machine Translation. *NAACL 7th Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-7)*, pp. 29–38, Atlanta, GA, USA, June 2013.

[Huck & Wuebker+ 11] M. Huck, J. Wuebker, C. Schmidt, M. Freitag, S. Peitz, D. Stein, A. Dagnelies, S. Mansour, G. Leusch, H. Ney: The RWTH Aachen Machine Translation System for WMT 2011. *EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, pp. 405–412, Edinburgh, UK, July 2011.

[IBM 54] IBM: 701 Translator. Press release, January 1954.

[Karakos & Eisner+ 08] D. Karakos, J. Eisner, S. Khudanpur, M. Dreyer: Machine translation system combination using ITG-based alignments. *46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies (ACL): Short Papers*, pp. 81–84, Columbus, OH, USA, June 2008.

[Koehn & Axelrod+ 05] P. Koehn, A. Axelrod, A. Birch, C. Callison-Burch, M. Osborne, D. Talbot, M. White: Edinburgh system description for the 2005 IWSLT speech translation evaluation. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 68–75, Pittsburgh, PA, USA, October 2005.

[Koehn & Hoang+ 07a] 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, E. Herbst: Moses: Open Source Toolkit for Statistical Machine Translation. *45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 177–180, Prague, Czech Republic, June 2007.

[Koehn & Hoang 07b] P. Koehn, H. Hoang: Factored Translation Models. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 868–876, Prague, Czech Republic, June 2007.

[Leusch & Freitag+ 11] G. Leusch, M. Freitag, H. Ney: The RWTH System Combination System for WMT 2011. *EMNLP 2011 Sixth Workshop on Statistical Machine Translation (WMT)*, pp. 152–158, Edinburgh, UK, July 2011.

[Leusch & Matusov+ 08] G. Leusch, E. Matusov, H. Ney: Complexity of finding the BLEU-optimal hypothesis in a confusion network. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 839–847, Honolulu, HI, USA, October 2008.

[Levenshtein 66] V.I. Levenshtein: Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, Vol. 10, pp. 707–710, February 1966.

[Lin & Och 04] C.Y. Lin, F.J. Och: Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. *The 42nd Annual Meeting on Association for Computational Linguistics (ACL)*, 605, Barcelona, Spain, July 2004.

[Matusov & Ueffing+ 06] E. Matusov, N. Ueffing, H. Ney: Computing Consensus Translation from Multiple Machine Translation Systems Using Enhanced Hypotheses Alignment. *Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 33–40, Trento, Italy, April 2006.

[Mohri 02] M. Mohri: Semiring Frameworks and Algorithms for Shortest-distance Problems. *J. Autom. Lang. Comb.*, Vol. 7, No. 3, pp. 321–350, January 2002.

[Niehues & Herrmann$^+$ 11] J. Niehues, T. Herrmann, S. Vogel, A. Waibel: Wider Context by Using Bilingual Language Models in Machine Translation. *Sixth Workshop on Statistical Machine Translation (WMT)*, pp. 198–206, Edinburgh, UK, July 2011.

[Niehues & Waibel 13] J. Niehues, A. Waibel: An MT Error-Driven Discriminative Word Lexicon using Sentence Structure Features. *Eighth Workshop on Statistical Machine Translation (WMT)*, pp. 512–520, Sofia, Bulgaria, August 2013.

[Nomoto 04] T. Nomoto: Multi-Engine Machine Translation with Voted Language Model. *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pp. 494–501, Barcelona, Spain, July 2004.

[Och 99] F.J. Och: An Efficient Method for Determining Bilingual Word Classes. *Ninth Conference on European Chapter of the Association for Computational Linguistics (EACL)*, pp. 71–76, Bergen, Norway, June 1999.

[Och 03] F.J. Och: Minimum Error Rate Training for Statistical Machine Translation. *41st Annual Meeting on Association for Computational Linguistics-Volume 1 (ACL)*, pp. 160–167, Barcelona, Spain, July 2003.

[Och & Ney 02] F.J. Och, H. Ney: Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. *40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 295–302, Philadelphia, Pennsylvania, USA, July 2002.

[Och & Ney 03] F.J. Och, H. Ney: A Systematic Comparison of Various Statistical Alignment Models. *Computational linguistics*, Vol. 29, No. 1, pp. 19–51, 2003.

[Papineni & Roukos$^+$ 02] K. Papineni, S. Roukos, T. Ward, W.J. Zhu: BLEU: a method for automatic evaluation of machine translation. *40th Annual Meeting on Association for Computational Linguistics (ACL)*, pp. 311–318, Philadelphia, PA, USA, July 2002.

[Peitz & Freitag$^+$ 11] S. Peitz, M. Freitag, A. Mauser, H. Ney: Modeling Punctuation Prediction as Machine Translation. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 238–245, San Francisco, CA, USA, December 2011.

[Peitz & Freitag$^+$ 14] S. Peitz, M. Freitag, H. Ney: Better Punctuation Prediction with Hierarchical Phrase-Based Translation. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 271–278, Lake Tahoe, CA, USA, December 2014.

[Peitz & Mansour$^+$ 12] S. Peitz, S. Mansour, M. Freitag, M. Feng, M. Huck, J. Wuebker, M. Nuhn, M. Nußbaum-Thom, H. Ney: The RWTH Aachen Speech Recognition and Machine Translation System for IWSLT 2012. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 69–76, Hong Kong, December 2012.

[Peitz & Mansour$^+$ 13a] S. Peitz, S. Mansour, M. Huck, M. Freitag, H. Ney, E. Cho, T. Herrmann, M. Mediani, J. Niehues, A. Waibel, A. Allauzen, Q.K. Do, B. Buschbeck, T. Wandmacher: Joint WMT 2013 Submission of the QUAERO Project. *Eighth Workshop on Statistical Machine Translation (WMT)*, pp. 185–192, Sofia, Bulgaria, August 2013.

[Peitz & Mansour$^+$ 13b] S. Peitz, S. Mansour, J.T. Peter, C. Schmidt, J. Wuebker, M. Huck, M. Freitag, H. Ney: The RWTH Aachen Machine Translation System for WMT 2013. *ACL 2013 Eighth Workshop on Statistical Machine Translation (WMT)*, pp. 193–199, Sofia, Bulgaria, August 2013.

[Peitz & Wuebker+ 14] S. Peitz, J. Wuebker, M. Freitag, H. Ney: The RWTH Aachen German-English Machine Translation System for WMT 2014. *ACL 2014 Ninth Workshop on Statistical Machine Translation (WMT)*, pp. 157–162, Baltimore, USA, June 2014.

[Rosti & Ayan+ 07] A.V. Rosti, N.F. Ayan, B. Xiang, S. Matsoukas, R. Schwartz, B. Dorr: Combining Outputs from Multiple Machine Translation Systems. *The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pp. 228–235, Rochester, NY, USA, April 2007.

[Rosti & He+ 12] A.V. Rosti, X. He, D. Karakos, G. Leusch, Y. Cao, M. Freitag, S. Matsoukas, H. Ney, J. Smith, B. Zhang: Review of Hypothesis Alignment Algorithms for MT System Combination via Confusion Network Decoding. *NAACL 2012 Seventh Workshop on Statistical Machine Translation (WMT)*, pp. 191–199, Montreal, Canada, June 2012.

[Rousseau 13] A. Rousseau: XenC: An Open-Source Tool for Data Selection in Natural Language Processing. *The Prague Bulletin of Mathematical Linguistics*, Vol. 100, No. 100, pp. 73–82, 2013.

[Schwenk & Gauvain 02] H. Schwenk, J.L. Gauvain: Connectionist Language Modeling for Large Vocabulary Continuous Speech Recognition. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 1, pp. I–765, Orlando, FL, USA, May 2002.

[Sim & Byrne+ 07] K.C. Sim, W.J. Byrne, M.J. Gales, H. Sahbi, P.C. Woodland: Consensus Network Decoding for Statistical Machine Translation System Combination. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 105–108, Honolulu, HI, USA, April 2007.

[Snover & Dorr+ 06] M. Snover, B. Dorr, R. Schwartz, L. Micciula, J. Makhoul: A Study of Translation Edit Rate with Targeted Human Annotation. *Association for Machine Translation in the Americas (AMTA)*, pp. 223–231, Cambridge, MA, USA, August 2006.

[Snover & Madnani+ 09] M. Snover, N. Madnani, B.J. Dorr, R. Schwartz: Fluency, adequacy, or HTER?: exploring different human judgments with a tunable MT metric. *Fourth Workshop on Statistical Machine Translation (WMT)*, pp. 259–268, Athens, Greece, March 2009. Association for Computational Linguistics.

[Son & Allauzen+ 12] L.H. Son, A. Allauzen, F. Yvon: Continuous Space Translation Models with Neural Networks. *The 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*, pp. 39–48, Montreal, Canada, June 2012.

[Stein & Vilar+ 11] D. Stein, D. Vilar, S. Peitz, M. Freitag, M. Huck, H. Ney: A Guide to Jane, an Open Source Hierarchical Translation Toolkit. *The Prague Bulletin of Mathematical Linguistics*, Vol. 95, pp. 5–18, April 2011.

[Stolcke 02] A. Stolcke: SRILM – An Extensible Language Modeling Toolkit. *International Conference on Speech and Language Processing (ICSLP)*, Vol. 2, pp. 901–904, Denver, CO, USA, September 2002.

[Sundermeyer & Alkhouli+ 14] M. Sundermeyer, T. Alkhouli, J. Wuebker, H. Ney: Translation Modeling with Bidirectional Recurrent Neural Networks. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 14–25, Doha, Qatar, October 2014.

[Vaswani & Zhao+ 13] A. Vaswani, Y. Zhao, V. Fossum, D. Chiang: Decoding with Large-Scale Neural Language Models Improves Translation. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1387–1392, Seattle, WA, USA, October 2013.

[Vilar & Stein+ 10] D. Vilar, D. Stein, M. Huck, H. Ney: Jane: Open Source Hierarchical Translation, Extended with Reordering and Lexicon Models. *ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR (WMT)*, pp. 262–270, Uppsala, Sweden, July 2010.

[Vogel 03] S. Vogel: SMT Decoder Dissected: Word Reordering. *International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE)*, pp. 561–566, Beijing, China, October 2003.

[Vogel & Ney+ 96] S. Vogel, H. Ney, C. Tillmann: HMM-Based Word Alignment in Statistical Translation. *16th Conference on Computational Linguistics (COLING)*, Vol. 2, pp. 836–841, Copenhagen, Denmark, August 1996.

[Watanabe & Sumita 02] T. Watanabe, E. Sumita: Bidirectional Decoding for Statistical Machine Translation. *International Conference on Computational Linguistics (COLING)*, pp. 1079–1085, Taipei, Taiwan, August 2002.

[Wu 97] D. Wu: Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. *Computational linguistics*, Vol. 23, No. 3, pp. 377–403, 1997.

[Wuebker & Huck+ 11] J. Wuebker, M. Huck, S. Mansour, M. Freitag, M. Feng, S. Peitz, C. Schmidt, H. Ney: The RWTH Aachen Machine Translation System for IWSLT 2011. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 106–113, San Francisco, CA, USA, December 2011.

[Wuebker & Huck+ 12] J. Wuebker, M. Huck, S. Peitz, M. Nuhn, M. Freitag, J.T. Peter, S. Mansour, H. Ney: Jane 2: Open Source Phrase-based and Hierarchical Statistical Machine Translation. *International Conference on Computational Linguistics (COLING)*, pp. 483–491, Mumbai, India, December 2012.

[Wuebker & Ney+ 12] J. Wuebker, H. Ney, R. Zens: Fast and Scalable Decoding with Language Model Look-Ahead for Phrase-based Statistical Machine Translation. *Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pp. 28–32, Jeju, Republic of Korea, July 2012.

[Wuebker & Peitz+ 13a] J. Wuebker, S. Peitz, T. Alkhouli, J.T. Peter, M. Feng, M. Freitag, H. Ney: The RWTH Aachen Machine Translation Systems for IWSLT 2013. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 88–93, Heidelberg, Germany, December 2013.

[Wuebker & Peitz+ 13b] J. Wuebker, S. Peitz, F. Rietig, H. Ney: Improving Statistical Machine Translation with Word Class Models. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1377–1381, Seattle, WA, USA, October 2013.

[Xiong & Zhang+ 11] D. Xiong, M. Zhang, H. Li: Enhancing Language Models in Statistical Machine Translation with Backward N-grams and Mutual Information Triggers. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 1288–1297, Portland, OR, USA, June 2011.

[Xiong & Zhang 15] D. Xiong, M. Zhang: Backward and trigger-based language models for statistical machine translation. *Natural Language Engineering*, Vol. 21, pp. 201–226, 3 2015.

[Zens & Ney$^+$ 04] R. Zens, H. Ney, T. Watanabe, E. Sumita: Reordering constraints for phrase-based statistical machine translation. *20th international conference on Computational Linguistics (COLING)*, pp. 205–2011, Geneva, Switzerland, August 2004. Association for Computational Linguistics.

[Zens & Ney 08] R. Zens, H. Ney: Improvements in Dynamic Programming Beam Search for Phrase-based Statistical Machine Translation. *International Workshop on Spoken Language Translation (IWSLT)*, pp. 195–205, Honolulu, HI, USA, October 2008.