Lost and Found in Translation: Cross-Lingual Question Answering with Result Translation

Kristen Parton

Submitted in partial fulfillment of the

requirements for the degree

of Doctor of Philosophy

in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2012

ABSTRACT

Lost and Found in Translation: Cross-Lingual Question Answering with Result Translation

Kristen Parton

Using cross-lingual question answering (CLQA), users can find information in languages that they do not know. In this thesis, we consider the broader problem of CLQA with result translation, where answers retrieved by a CLQA system must be translated back to the user's language by a machine translation (MT) system. This task is challenging because answers must be both relevant to the question and adequately translated in order to be correct. In this work, we show that integrating the MT closely with cross-lingual retrieval can improve result relevance and we further demonstrate that automatically correcting errors in the MT output can improve the adequacy of translated results.

To understand the task better, we undertake detailed error analyses examining the impact of MT errors on CLQA with result translation. We identify which MT errors are most detrimental to the task and how different cross-lingual information retrieval (CLIR) systems respond to different kinds of MT errors. We describe two main types of CLQA errors caused by MT errors: lost in retrieval errors, where relevant results are not returned, and lost in translation errors, where relevant results are perceived irrelevant due to inadequate MT.

To address the lost in retrieval errors, we introduce two novel models for cross-lingual information retrieval that combine complementary source-language and target-language information from MT. We show empirically that these hybrid, bilingual models outperform both monolingual models and a prior hybrid model.

Even once relevant results are retrieved, if they are not translated adequately, users will not understand that they are relevant. Rather than improving a specific MT system, we take a more general approach that can be applied to the output of any MT system. Our adequacy-oriented automatic post-editors (APEs) use resources from the CLQA context and information from the MT system to automatically detect and correct phrase-level errors in MT at query time, focusing on the errors that are most likely to impact CLQA: deleted or missing content words and mistranslated named entities. Human evaluations show that these adequacy-oriented APEs can successfully adapt task-agnostic MT systems to the needs of the CLQA task.

Since there is no existing test data for translingual QA or IR tasks, we create a translingual information retrieval (TLIR) evaluation corpus. Furthermore, we develop an analysis framework for isolating the impact of MT errors on CLIR and on result understanding, as well as evaluating the whole TLIR task. We use the TLIR corpus to carry out a task-embedded MT evaluation, which shows that our CLIR models address lost in retrieval errors, resulting in higher TLIR recall; and that the APEs successfully correct many lost in translation errors, leading to more adequately translated results.

Table of Contents

Acronyms

APE	automatic post-editor
CLIA	cross-lingual information access
CLIR	cross-lingual information retrieval
CLQA	cross-lingual question answering
DT	document translation
MAP	mean average precision
MT	machine translation
NDCG	normalized discounted cumulative gain
NE	named entity
OOV	out of vocabulary
POS	part of speech
QT	query translation
SMLIR	simultaneous multilingual information retrieval
SMT	statistical machine translation
TLIR	translingual information retrieval
TLQA	translingual question answering

List of Figures

List of Tables

List of Algorithms

Chapter 1

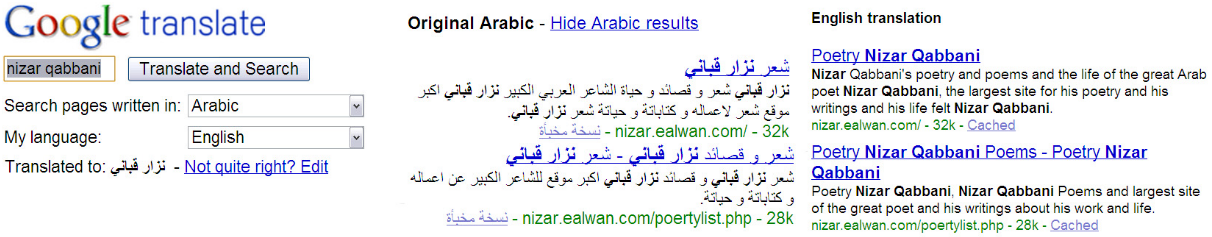
Introduction

The growth of the World Wide Web over the past 20 years has given us unprecedented access to content in different languages. With a few clicks, any user can read the latest news from Egypt in Arabic, follow the latest tweets in Japanese from the Japanese Prime Minister or watch the latest episode of a popular Mexican soap opera in Spanish. With the increasing quality and availability of machine translation (MT) on the Internet, users can consume content in many languages other than their own.

Cross-lingual applications go beyond passive consumption to enable users to find, analyze and use information in other languages. For instance, an English-speaking user can find Arabic news articles about *Mohamed Tantawi*, Japanese tweets about the *Japan earthquake recovery* or a list in Spanish of *the most popular telenovelas*.

Cross-lingual applications and MT are two sides of the same coin: the cross-lingual application satisfies a user's information need in another language and MT makes the information understandable to the user in his or her own language. Without a cross-lingual application, users can only browse and read foreign-language information, rather than search and analyze it. Without MT, users can find information relevant to their needs, but cannot read it because it is in another language.

In this thesis, we examine the special case of cross-lingual applications with result translation, focusing on the cross-lingual question answering (CLQA) task and the related cross-lingual information retrieval (CLIR) task. Figure 1.1 shows an example of CLIR with result translation: the user enters a query in English, the CLIR system returns a set of documents in Arabic, and then


Google translate
nizar qabbani
Translate and Search
Search pages written in: Arabic
My language: English
Translated to: نزار قباني - Not quite right? Edit
Original Arabic - Hide Arabic results
English translation

شعر نزار قباني
نزار قباني شعر و قصائد و حياة الشاعر العربي الكبير نزار قباني اكبر موقع شعر لاعماله و كتاباتة و حياتة شعر نزار قباني.
نسخة مخبأة - nizar.ealwan.com/ - 32k

شعر و قصائد نزار قباني - شعر نزار قباني
شعر نزار قباني و قصائد نزار قباني اكبر موقع للشاعر الكبير عن اعماله و كتاباتة و حياتة.
نسخة مخبأة - nizar.ealwan.com/poertylist.php - 28k

Poetry Nizar Qabbani
Nizar Qabbani's poetry and poems and the life of the great Arab poet Nizar Qabbani, the largest site for his poetry and his writings and his life felt Nizar Qabbani.
nizar.ealwan.com/ - 32k - Cached

Poetry Nizar Qabbani Poems - Poetry Nizar Qabbani
Poetry Nizar Qabbani, Nizar Qabbani Poems and largest site of the great poet and his writings about his work and life.
nizar.ealwan.com/poertylist.php - 28k - Cached

Figure 1.1: A commercial example of CLIR with result translation (by Google). Traditional CLIR systems would only show the Arabic results.

the MT system translates them back into English. Given a set of documents in language L_d and an open-ended question in language L_q, the goal of CLQA is to return a set of answers in language L_d that are relevant to the question. In CLQA with result translation, the answers must be machine translated back into the question language, L_q, so the user can read them. Since MT is embedded within the task definition and evaluation, we refer to it as **task-embedded MT**. Alternatively, we refer to any application that makes a round-trip "through" another language and back to the original language as a **translingual** task, in contrast with cross-lingual tasks, which only go "across" the language gap once.

Task-embedded MT spans two mature research fields: cross-lingual information access and machine translation. Cross-lingual information access covers a wide range of tasks that enable users to access information in languages other than their own, including information retrieval, question answering, information extraction and summarization. Typically in cross-lingual applications, the results are returned in the document language rather than the user language, under the assumption that MT is a separate, post-processing step. On the other hand, most research in MT is concerned with application-agnostic, open-domain translation.

Although these two fields seem naturally symbiotic, there are several reasons that task-embedded MT has not been widely studied in the past. Developing a full end-to-end cross-lingual system with result translation is very resource-intensive, since it requires a cross-lingual application, one or more MT systems and a way to integrate them. Doing task evaluation on machine translated output is difficult and it is not always possible to get high inter-annotator agreement. Furthermore, MT systems evolve quickly, so any algorithms developed for adapting MT output to the task must be general enough to apply to newer and better MT systems.

Despite these challenges, integrating these two areas in task-embedded MT provides many significant benefits. Considering MT in the context of a task enables extrinsic evaluations of MT, where the *usability* of a particular MT system can be assessed, rather than just intrinsic quality. The task context provides specific desiderata for the MT output, which means that the MT can be tailored to the needs of the task. Furthermore, by using bilingual information from the source language and MT, task performance can also be improved. Finally, task-embedded MT models real-world, end-to-end applications, where results must be translated so users can understand them.

1.1 Approach

The research in this thesis bridges the gap between cross-lingual applications and machine translation by exploring them both in the context of *task-embedded MT*. This thesis delves deeply into the complex interactions between the two that lead to compounding errors, as well as the opportunities presented for improving performance of one by using information available in the other.

1.1.1 Task-Embedded MT

Task-embedded MT: Any multi-lingual or cross-lingual natural language processing (NLP) task that is evaluated on MT output.

There are several components to this definition. The definition is general enough to cover any *NLP task*, that is, a text processing problem with well-defined output that can be judged correct or incorrect, such as parsing, information retrieval (IR), information extraction, sentiment analysis, question answering (QA) or textual entailment. The task must also be *cross-lingual*. It is easy to see how tasks such as IR or QA are defined in a cross-lingual context: the user enters a query or question in one language and the corpus is in another language. Other NLP tasks may be cross-lingual without a user query. For example, cross-lingual semantic role labeling is defined as identifying semantic roles in a translated sentence.

The final aspect of task-embedded MT is that the task evaluation must be done on translated results. This means that overall system performance must be based on (machine) translated output, rather than judging the output of the system in the document source language. The latter requirement is crucial because it ties system performance to both task performance and machine translation quality, and this is what distinguishes task-embedded MT from other cross-lingual tasks.

1.1.2 CLQA with Result Translation as Task-Embedded MT

CLQA with result translation clearly fits the definition of task-embedded MT. This holistic approach requires us to view CLQA and MT from novel perspectives, and challenges traditional assumptions about each. In CLIR, is it useful to retrieve a relevant result if the result translation cannot be understood by the user? In MT, if there is a trade-off between preserving meaning and producing a grammatical translated result, are we willing to lose some information in order to gain fluency? How can we intelligently combine CLQA and MT to optimize the end-to-end translated relevance? We consider all of these questions in this thesis.

In CLQA with result translation, the MT system and CLQA system each have an impact on the other. Since many CLQA models rely on MT to find answers, MT errors can degrade answer retrieval as well as answer translation. In other words, in the end-to-end system, the MT system can impact how well a result is translated and also whether the result is found in the first place. While CLQA systems are typically responsible for retrieving and ranking results in the document language, task-aware CLQA systems can also take into account result translations during ranking. If the CLQA system ranks relevant results with good translations higher than relevant results with bad translations, it can impact both retrieval as well as how well the top-k result translations are understood.

Since the CLQA and MT systems are interdependent, errors in one component can have a compounding effect on the end-to-end system. To understand the task better, we undertake detailed error analyses examining the impact of MT errors on CLQA with result translation. We identify which MT errors are most detrimental to the task and how different CLIR systems respond to different kinds of MT errors. We study different MT systems to ensure that our results are not due to the idiosyncrasies of a particular system or approach.

Based on the error analysis results, we develop novel CLIR models that take into account result translation. The models are particularly suitable for the translingual IR task, and they also improve over previous models in the CLIR task (with no result translation). In this case, improvements designed for the extrinsic evaluation of the CLIR system resulted in intrinsic gain as well.

We also explore different ways to improve the MT output given the CLQA context. We specifically focus on the MT errors that were found to be most harmful to CLQA relevance during our error analysis. Rather than improving a specific MT system, we take a more general approach that can be applied to the output of any MT system. These automatic post-editors use resources from the CLQA context and information from the MT system to detect and correct errors in result translations. Again, this approach was specifically designed for task-embedded MT, but intrinsic evaluations show that it also helps improve MT adequacy in general.

After first considering the task from the perspective of CLQA and then from the perspective of MT, we finally evaluate an end-to-end system. In order to do so, we had to build a new evaluation corpus specifically targeting translingual information retrieval (TLIR). This extrinsic corpus yielded results that previous intrinsic evaluations could not address. Even for our baseline MT and CLIR systems, intrinsic results did not always match the extrinsic results, so the extrinsic TLIR evaluation was an important complement to intrinsic evaluations of CLIR retrieval and MT quality.

Using the TLIR corpus, we can measure the impact of the novel CLIR models and the MT post-editors on the end-to-end system. The task-aware CLIR models focus on finding results that were previously lost (unretrieved) due to MT errors. The post-editors aim to restore information in machine translated sentences where the meaning had previous been lost in translation. Both the MT and CLIR systems are able to leverage the task context to improve end-to-end performance.

1.2 Overview of Thesis Contributions

The contributions of this thesis are:

Task-Based MT Error Analysis: We present the results of detailed error analyses of output from three different MT systems. We examined only errors that would directly impact performance on the CLQA task. Our analyses showed that MT adequacy was degraded by mistranslated named entities and deleted or mistranslated content words. These types of errors were present in all the corpora and MT systems that we examined, even as we analyzed newer and better MT systems.

Novel Bilingual Models for CLIR: We introduce Simultaneous Multilingual Information Retrieval (SMLIR), a novel hybrid model for CLIR that combines the complementary advantages of two standard CLIR models, query translation and document translation. The SMLIR model is particularly well suited for task-embedded MT because it assumes that the CLIR results must be translated back to the query language so the user can read them. We present results from a Chinese-English document-level CLIR evaluation showing that SMLIR outperforms both previous CLIR models on cross-lingual retrieval (with no result translation). We also describe a separate evaluation on an Arabic-English sentence-level CLIR task, comparing the models on both cross-lingual and translingual retrieval, where the latter includes result translation. In both cases, SMLIR performed better than the baseline CLIR models. We also present a less resource-intensive model, query translation re-rank (QT-rerank), that performs as well as SMLIR.

Methods for Detecting and Correcting MT Errors: The MT error analyses motivated the development of targeted error detection and correction algorithms for MT. We present an algorithm for detecting phrase-level adequacy errors in MT with high precision. We describe several different techniques for automatically post-editing the detected errors: a rule-based automatic post-editor (APE) and two types of feedback APEs. Since the APEs are general, they can be applied to any MT system that produces word alignments. An in-depth MT evaluation showed that these APEs reduced errors in MT adequacy, which is crucial for both CLIR and CLQA.

TLIR Evaluation Corpus: Designing and carrying out an evaluation of task-embedded MT is challenging due to interactions between system components and compounding errors. We present a testbed for evaluating CLIR with result translation. This TLIR evaluation corpus has manual relevance judgments on gold translations and two different machine translations across two different genres.

Task-Embedded MT Evaluation: By comparing judgments on gold translations and machine translations, we quantify the impact of MT errors on two components of the TLIR task, retrieval accuracy and result understanding, as well as on the entire end-to-end TLIR system. Finally, we present an extrinsic evaluation of our novel CLIR models and APEs, which was made possible by the TLIR evaluation corpus.

1.3 Outline of Thesis

This thesis is structured as follows. First, we give background information on the separate fields of MT and cross-lingual information access and how each of them relates to task-embedded MT (Chapter 2). Then we present error analyses of three different MT systems from the perspective of the CLQA task, and also describe the MT systems that will be used in experiments throughout the thesis (Chapter 3).

Then we consider task-embedded MT from the perspectives of CLQA and MT separately. We introduce the SMLIR model (Chapter 4), which was motivated by the errors described in our error analysis. Evaluations on the traditional CLIR task (without result translation) show that this hybrid, bilingual model outperforms monolingual models as well as previous hybrid models. In other words, by taking into account the translated results, the model is able to retrieve more relevant documents.

Next we explore task-embedded MT from the MT viewpoint. Once relevant documents are retrieved, MT errors in the result translations can still make a relevant document *appear* irrelevant. We present several techniques for detecting errors in MT adequacy (Chapter 5), focusing on the specific types of problems that were found to be most detrimental to task performance in the error analyses. Once MT errors are detected, we present and compare multiple approaches for automatically correcting the detected MT errors (Chapter 6).

The experiments in Chapters 4 and 6 are based on intrinsic evaluations of CLIR and MT, respectively. Chapter 7 takes a holistic view of task-embedded MT and presents task-based evaluations. We describe the TLIR evaluation corpus and experiments on our CLIR and APE models using the TLIR evaluation corpus. Finally, we summarize our contributions and discuss directions for future work (Chapter 8).

This thesis spans two fields, MT and CLIR, and uses technical terms from both fields, including many acronyms. For easier reading, we re-define unfamiliar acronyms on their first use in each chapter. We have also included a glossary (page 207) that defines important terms and spells out all the acronyms, as well as a simple acronym list (page vii).

Chapter 2

Background

Task-embedded MT spans the nearly disjoint fields of MT and cross-lingual information access, and is meant to be of interest to researchers in one field who have no expertise in the other. For MT researchers, it provides an extrinsic evaluation of MT output, and for researchers in cross-lingual information access, it models a real-world application, where the user can actually read the results of a CLQA or CLIR system. In this chapter, we provide a high-level overview of these two separate fields and how they relate to task-embedded MT.

There are complete textbooks written on each of these topics (e.g., [Koehn, 2010] and [Nie, 2010]), so this chapter is not meant to be a comprehensive tutorial on each of them, but rather a frame of reference for understanding our motivation and the challenges we face. Our work in this thesis is meant to be general enough to apply to any MT system and a variety of cross-lingual tasks, so we limit this chapter to a high-level discussion of these two fields, and leave implementation details to the experimental sections in later chapters.

2.1 Overview of MT

Every day, Google MT translates “as much text as you’d find in 1 million books” [Och, 2012] (as of April 2012). The ubiquitous need for MT will only increase as Internet penetration grows in more countries, bringing with it more multilingual content in a wider range of diverse languages and more contact between speakers of different languages. While initial work on MT in the 1950’s (in the US) was motivated by the needs of the defense industry, and MT progressed over time due to localization needs of businesses, the Internet has made the need for MT more personal. Today MT is needed to understand a friend's Facebook status, a colleague's tweet, or a grandmother's email.

Along with the growing need for MT comes additional challenges for MT. Currently, translating news articles from French to English can be done with fairly high quality and consistency on a cluster of computers. However, translating from a very different genre, such as Twitter, is much harder: the language is less formal, there are many misspellings and abbreviations, and there is very little training data in the form of human-translated tweets. Similarly, translating to or from a "minority" or "low-resource" language is difficult because there is very little training data and few basic tools for parsing or processing the data. Communicating with one's grandmother in Ladino or Warlpiri via MT is not currently possible (although MT systems for Yiddish and Welsh are available!). A further challenge is to do open-domain MT on a mobile device with limited processor power and memory that is not connected to the Internet. Currently, several companies offer speech-to-speech translation systems on smart phones, but they rely on a data connection and ultimately still use a large computing cluster to process the data. The military has small translation devices that are not tethered to a data connection, but they are typically limited to closed-domain MT (as far as we know) such as handling conversations at road checkpoints.

This means that even as MT quality gets better with increasing amounts of training data, faster computers and smarter translation models, MT will remain a hard problem for years to come. In other words, even when MT systems get better, task-embedded MT will need to be able to handle errors in MT.

In the rest of this section, we define MT, give an overview of several types of MT models, and finally discuss how to evaluate MT output.

2.1.1 Definition: What is MT?

A **machine translation system** takes input in source language L and automatically produces output in target language M, where the output is adequate and fluent.

The **input** and **output** may be any form of natural language - written, spoken or even sign language - but in this thesis we are concerned only with written or transcribed text. The text may be in any format - a newspaper article, a web page, closed captions for a TV show, a dialogue, a blog, a text message or a tweet.

The terms **source language** and **target language** are used to refer to the input and output languages, respectively. (Although in MT papers using the noisy channel model, the definitions of these terms are reversed.) We assume that a particular MT system produces translations between two specific languages in one direction (i.e., two separate MT systems are needed for English-to-French and French-to-English), rather than acting as a universal translator from any language to any other language. Many MT tools support multiple language pairs – for instance, Google translate currently handles translation between 64 languages in either direction (4,032 language pairs!) – but that is not the focus of this thesis.

By **automatically**, we mean that the system produces translations with no human assistance or intervention. In contrast, computer-aided translation refers to translations produced by human translators who use software to assist in translation. In its simplest form, this may simply consist of a human translator using an electronic dictionary or spell checker. At the other end of the spectrum, interactive MT is an MT system with a human in the loop to edit and correct the translations produced by a computer. In this thesis, we are only interested in fully automatic MT systems.

An **adequate** translation is one that carries the same meaning as the original text. In other words, adequacy is a measure of how faithful the translation is to the original. In translation theory, adequacy is also referred to as fidelity or faithfulness.

A **fluent** translation is one that is grammatical and understandable by a native speaker of target language M. Fluency is sometimes referred to as transparency, and fluent translations can also be called idiomatic. While translation adequacy is measured with respect to the source-language input sentence, fluency is independent of the original sentence and relies only on the target-language output sentence.

2.1.2 Statistical MT: How does MT work?

Much of our work in this thesis involves analyzing the impact of MT errors on CLQA and proposing methods to handle them. In order to understand why and how MT errors arise, it helps to understand how MT systems work.

Research in MT has followed the same trends as research in artificial intelligence. Initially, MT systems were expert systems built by computational linguists. These *rule-based MT* systems consisted of hand-built translation rules between two languages. As digital translation data became available, statistical models were built to learn these translation rules using machine learning. As machine learning techniques became more sophisticated and the amount of available data increased exponentially, these *statistical machine translation (SMT)* models became the state-of-the-art in MT. The industry leader in rule-based MT, Systran, now incorporates SMT models into their rule-based MT systems [Systran, 2012].

SMT models are built using two primary sources of training data: parallel corpora and monolingual corpora. Parallel corpora are existing translations produced by humans in two or more languages. For example, official documents from the European Parliament are produced in all 23 official languages of the European Union. Many websites are translated into multiple languages, so the web is an excellent source of parallel data in the non-governmental domain [Uszkoreit *et al.*, 2010; Resnik and Smith, 2003]. SMT systems also use large amounts of monolingual data in the target (output) language. These corpora are much easier to find than parallel corpora; for instance, the LDC currently produces Gigaword in 5 languages, and the Google Web 1T 5-gram corpus is available in 10 European languages (LDC2009T25).

Parallel data is used to train **translation models** that map from the source language to the target language, while the monolingual data is used to build a **language model (LM)** to estimate how likely different translations are in the target language. These two different components of an SMT system correspond very roughly to the concepts of adequacy and fluency: the translation model is responsible for adequacy, while the language model estimates fluency.

Both the translation model and the LM can work on different representations of the input and output sentences. (Even when the input is a large document, most MT systems translate each sentence in a document independently.) At the shallowest level, a sentence is simply a string of tokens. Deeper levels of representation contain additional layers of linguistic annotation: part-of-speech tags, morphological analysis, syntactic parses, and semantic parses. Linguistic annotations are produced automatically by a parser, which requires its own training data. LMs are typically based on shallow representations, such as token n-grams, which may be used in conjunction with part-of-speech LMs. Translation models range widely, and the name of a particular SMT system usually refers to the type of translation model it uses. The following list briefly describes the most popular SMT translation models from simpler to more complex, including those that we use in our experiments.

- **Word-based**: Unigrams are mapped to unigrams. The parent of modern SMT models, the IBM model 1 [Brown *et al.*, 1993], is a word-to-word translation model that also allows insertion and deletion.

- **Phrase-based**: N-grams are mapped to n-grams. A phrase-based model includes unigram rules, so it is a superset of a word-based model. The source-language and target-language n-grams do not have to be the same length. The term "phrase" does not indicate that the n-grams are coherent linguistic units; e.g., ", to the" is an example of a trigram phrase. The popular Moses open-source SMT toolkit [Koehn *et al.*, 2007] implements a phrase-based model. By capturing translations of n-grams instead of single tokens, phrase-based models exploit context to reduce translation ambiguity. For example, the English word "bank" can be translated into Spanish as "banco" (financial institution), "orilla" (bank of a river), or "contar" (to count on), among others. However, when translating the n-grams "investment bank", "bank of a river", or "to bank on", the lexical choice for "bank" is unambiguous.

- **Hierarchical phrase-based**: Trees are mapped to trees, but the trees are not linguistic units. The model learns a synchronous context-free grammar (SCFG), where the grammar rules are composed of arbitrary n-grams and non-terminals. (For example, a Spanish-English translation rule could map "no me V" to "don't V me", where the V non-terminals would be filled by another phrase translation.) Neither the n-grams nor the non-terminals are constrained to map to linguistically motivated phrases. These models were introduced by [Chiang, 2005]. This model is a generalization of the phrase-based model, since rules without non-terminals are simply phrase translation rules. The main advantage of hierarchical rules is that they allow long-distance re-ordering of phrases, whereas phrase-based SMT systems can typically only re-order phrases within a fixed window. This is particularly important when translating between languages with different word orders, for instance Arabic to English, where sentences with the Arabic word order verb-subject-object (VSO) must be translated into the English word order subject-verb-object (SVO).

- **Syntax-based**: Linguistically motivated trees are mapped to linguistically motivated trees.

> This model is also an SCFG, but the grammar is based on syntactically well-formed trees. Non-terminals must map to linguistically motivated phrases (such as noun phrases or verb phrases) rather than arbitrary n-grams. The motivation for syntax-based models is that they should produce more fluent output than hierarchical models, because the rules are constrained to match syntactically well-formed trees only. Another advantage is that the translation rules are easier for humans to understand, which can help if a human is involved in post-editing or correcting translations. However, purely syntax-based models do not consistently outperform hierarchical models, and improving syntactic translation models is an active area of research. Both hierarchical and syntax-based models also come in asymmetric versions, where trees are used only on the source side (tree-to-string) or only on the target side (string-to-tree).

There is a trade-off between using a shallower, faster model (such as a phrase-based model) and using a more powerful, slower model with exponentially more parameters (such as a syntax-based model). Models based on shallow representations have no linguistic knowledge, so are necessarily limited: they over-generate sentences that are linguistically ill-formed, yet are unable to carry out many linguistically necessary transformations, such as long-distance movement. Deeper representations can provide the MT system with additional linguistic knowledge, but since they have more parameters, their coverage is much sparser given the same amount of training data. If the model has too many parameters, translation can become computationally intractable. Furthermore, the accuracy of a model built on a complex representation is dependent on the accuracy of the parser, which varies greatly by language. (In a recent shared parsing task, the best English parser was significantly more accurate than the best Arabic parser (90% versus 77%) [Nivre *et al.*, 2007].) Much current research in SMT models is aimed at incorporating more linguistic knowledge into MT systems, given limitations on the amount of parallel text available and constraints on translation runtime.

In addition to the translation model and the LM, SMT systems may have an arbitrary number of additional parameters or models. For instance, many phrase-based SMT systems have distortion parameters that control how phrases are re-ordered during translation, whereas syntax-based SMT systems handle re-ordering directly in the translation model.

During training, the model parameters are estimated from the training data. In the tuning step, a small development set is used to set values for meta-parameters, for example the relative weighting of the translation model and the LM. At translation time, the SMT system uses the models to "decode" the source-language input sentence into a target-language output translation. Each word or phrase in the output sentence is derived from a word or phrase in the source sentence. word or phrase alignmentsWord (or phrase) alignments represent the mapping from source-language input words to target-language output words that was used by the decoder to produce the translation. Depending on the model used, the alignments may be word-to-word, phrase-to-phrase or based on translation rules.

The algorithms used for training, tuning and decoding depend on the types of models used and the SMT system implementation. The rapid pace of research in SMT ensures that the algorithms are constantly evolving and improving.

2.1.3 MT Evaluation: How well does MT work?

A popular adage says that translations can either be faithful or beautiful, but not both. Even human translators have difficulty producing translations that are both fluent and adequate, as the second and fourth examples in Table 2.1 show. If trained professionals make mistakes, it is not surprising that MT systems also make fluency and adequacy errors, particularly since SMT systems are trained on human-translated data.

Translation is fundamentally different from many other problems in natural language processing or machine learning because there is no single correct answer. A sentence has one correct syntactic parse and a stream of English speech has a single correct transcription, but a French sentence may be expressed in English in innumerable different ways. Scholars have been producing English translations of Homer's Odyssey for centuries - Wikipedia currently lists 24 different translations. Each one is different, yet they are all acceptable translations.

Even determining the accuracy of a translated sentence is a challenge. In question answering, a given question may have multiple possible answers, but given one answer, it is generally easy to determine whether it is correct. Similarly, a web query have many relevant pages on the Internet, but given a single result, humans can rate its relevance to a query with a high degree of inter-annotator agreement (which measures agreement between different annotators). In contrast, when humans are asked to rate translations, they have only fair inter-annotator agreement, and only moderate intra-annotator agreement (which measures how well annotators agree with their own

Source	Translation / Reference / *Data source*	Fluent?	Adequate?	
MT	And the founder of WikiLeaks, Julian Assange, who is wanted by Interpol, is probably in Britain, according to information published by newspapers.	Yes	Yes	
Ref.	In the meantime, a newspaper report suggests Wikileaks founder Julian Assange, who is on Interpol's wanted list, is currently in Great Britain.			
	WMT11 French-English			
Human	Foreign experts estimate that Israel possesses, due to this reactor amounts of plutonium enough, from between 100 to 200 nuclear heads, for long-range missiles.	No	Yes	
Ref.	Foreign experts estimate that, thanks to this reactor, Israel has sufficient plutonium to arm between 100 and 200 nuclear warheads for long-range missiles.			
	NIST02 Arabic-English			
MT	He said that in twenty minutes.	Yes	No	
Ref.	Jancura announced this on the Twenty Minutes programme on Radiozurnal.			
	WMT10 Czech-English			
Human	At the age of 46, with a past medical carrier, he was charged of <<operate>> on that <<wounded>> solar array.	No	No	
Ref.	At the age of 46, this doctor by training was responsible for "operating" on the "wounded" solar array.			
	WMT09 French-English			

Table 2.1: Examples of translations covering all combinations of fluent/not fluent and adequate/not adequate, from a range of MT development and test sets. Note that even translations produced by humans can be disfluent and/or inadequate, and yet we rely on them to train, tune and test our MT systems.

judgments) [Callison-Burch *et al.*, 2007].

How to evaluate translation has been an open question since the early days of “mechanical translation.” A half-century old paper Miller and Beebe-Center [1956] outlines ideas for MT evaluation that mirror the current major trends in MT evaluation:

1. **Manual evaluation**: “One can ask the opinion of several competent judges.”

2. **Automatic metrics**: “One can compare [human] translations with [MT] by a variety of statistical indices.”

3. **Task-based evaluation**: “Or a person who has read only the translation may be required to answer questions based on the original.”

The experiments in this thesis use all of these methods, though ultimately we are most interested in task-based evaluation of MT, where the task is CLQA.

Manual evaluation of MT output can be carried out in many ways, several of which are shown in Table 2.2. If the annotators are bilingual, they can compare the translations to the source text, but since bilingual annotators are more expensive, many manual evaluations are done using monolingual annotators. In these evaluations, the machine translated sentence is compared to one or more **reference translations**, which are human translations of the same sentence. The annotator may be asked to rate a single translated sentence, or to rank multiple translations of the same sentence relative to each other. Rating translations is typically done using 5-point adequacy and fluency scales, as shown in Table 2.2, but even when the scales are well-defined, the ratings tend to be subjective (though [Denkowski and Lavie, 2010] describe a sophisticated method for normalizing ratings). Pairwise ranking of translated sentences (also shown in Table 2.2) has higher agreement [Callison-Burch *et al.*, 2007], but since all of the annotations are relative to specific MT systems, the results are not easily comparable across different evaluations. A cheaper, more reusable type of annotation is a binary “acceptable or not acceptable” judgment for each translation (the “Acc.?” column in Table 2.2), though this coarse-grained decision does not distinguish between translations that are almost acceptable and those that are incomprehensible.

Automatic metrics seek to avoid the cost associated with manual evaluation by automatically scoring translation output. Better automatic metrics will have higher correlation with human

Sys.	Translation / Comments	Flu. (1-5)	Adeq. (1-5)	Acc.? (Y/N)	Rank (1-5)
Ref.	The American president's envoy for Sudan, Andrew Natsios, started his visit to the country yesterday.	5	5	Yes	1
	Completely adequate and fluent.				
MT A	Launched a U.S. presidential envoy to Sudan Andrew Natsios visit to the country yesterday.	2	4	Yes	2
	"Launched" was not re-ordered correctly, and is an awkward lexical choice.				
MT B	US President started envoy to Sudan Andrew natsios visit to the country.	1	2	No	4
	"Started" is in the wrong place and changes the meaning of the noun phrase "President's envoy." The word "yesterday" is missing.				
MT C	The American president's envoy to Sudan Andrew natsios visit to the country yesterday.	2	3	No	3
	The verb "started" is missing.				

Table 2.2: Different ways of manually evaluating MT output: 5-point fluency (Flu.) and adequacy (Adeq.) scales, binary acceptability (Acc.), and relative rank. Ref. is a human translation. MT A is an online SMT system; MT B uses a phrase-based model; and MT B uses a hierarchical phrase-based model.

judgments of MT. Most automatic metrics compare a machine translated corpus to one or more reference translations and produce a score representing how similar the MT is to the human translations.

A variety of intrinsic measures of MT quality have been proposed, but by far the most influential automatic metric has been BLEU (bilingual evaluation understudy) [Papineni *et al.*, 2002b]. The score calculates the modified n-gram precision for n=1 to N (usually N=4), and then takes the geometric mean, and finally multiplies by a brevity penalty to discourage cheating. BLEU was designed to satisfy the "need for quick evaluations" during system development, but it soon became the standard way for evaluating MT system quality. It has high system-level correlations with human judgments, yet is cheaper than human judgments (once the references are created) and is also reusable.

Since human ratings are inconsistent, it is not clear that measuring metrics by how well they correlate with humans is enough. Callison-Burch *et al.* [2006] demonstrate that increased BLEU score is "neither necessary nor sufficient for achieving an actual improvement in translation quality" and discuss specific cases where BLEU does not correlate with humans. Another problem with BLEU is that it is purely precision-based, so recall is not taken into account, which is a major problem for the tasks we address. In other words, words that are missing in the translation output do not always reduce BLEU score. Furthermore, there is no flexible unigram matching (for synonyms, spelling variations, etc.). So if the MT system produces a phrase that means the same thing but does not exactly match the reference translations (e.g., "pretty" instead of "beautiful"), it is counted as incorrect. Finally, a critical shortcoming of BLEU that we discuss further in later chapters is that all tokens are treated equally, so a comma is just as important as a verb. Many metrics have been suggested to remedy these problems: e.g., METEOR (metric for evaluation of translation with explicit ordering) [Lavie and Agarwal, 2007], HTER [Snover *et al.*, 2006], TERp (translation edit rate plus) [Snover *et al.*, 2009], syntax-based [Owczarzak *et al.*, 2007], and meta-metrics [Giménez and Màrquez, 2008]. BLEU remains the de facto standard for reporting MT improvements in research papers, and is still the "official primary metric" for the 2012 NIST OpenMT shared task.

The argument for **task-based MT evaluation** is that, since intrinsic evaluation is so difficult, task performance can be a proxy. Jones *et al.* [2007] created comprehension questions based on

Arabic documents, and then gave the questions to humans along with the documents translated into English. Not surprisingly, humans did much better on reference translations than MT. More interestingly, the results of the comprehension test did not always correlate with intrinsic measures of MT quality. Humans can sometimes discern important information even from garbled translations; on the other hand, a high-precision match to a reference translation could still be missing information that is crucial for answering a reading comprehension question. Task-based evaluation is complementary to intrinsic evaluation because it provides insight into MT usability.

While intrinsic metrics such as BLEU and METEOR seek to measure MT quality, they say very little about *usability* of MT output. In 1972-73, a user study was done on a low quality Russian-English MT system: it took twice as long for a human to read the MT output as native English text, and 21% of the text produced was completely unintelligible. To the researchers' great surprise, 96% of the users would recommend the system to their colleagues [Church and Hovy, 1993]. This apparent contradiction between the quality of the MT system and the usability of the MT system underscores the need to consider task context when evaluating the usability of MT systems.

2.2 Overview of CLIA

In this thesis, we explore task-embedded MT within the context of cross-lingual information access (CLIA), specifically the tasks of cross-lingual information retrieval (CLIR) and cross-lingual question answering (CLQA). CLIR enables a user to search documents written in languages that the user cannot read. This may seem counter-intuitive: why would the user want search results that he cannot understand? There are some situations where CLIR alone is sufficient.

One case is where users can understand the results, at least partially, but cannot search. The number of English second-language speakers worldwide exceeds the number of native English speakers, and despite the growing linguistic diversity of the web, there is still a great deal of information online that is available only in English. Second-language speakers who are comfortable reading documents in the second language may not be proficient enough to come up with good search terms, so CLIR alone is helpful.

Even without machine translation of the results, CLIR can be useful for filtering or selecting relevant pieces of information from a large corpus so that they can later be translated by humans or processed by native speakers of that language. Patent search is an example domain where CLIR can be helpful, and shared tasks in cross-lingual patent search have been run by NTCIR [Fujii *et al.*, 2007]. For example, CLIR can be an important first step for English-speaking patent attorneys who wish to search for prior art in the corpus of Chinese patents. As Fujii *et al.* [2007] describe, a basic "technology survey" search provides information about how many relevant patents exist, while an advanced "invalidity" search takes as input an existing patent and seeks to find exactly those "prior art" patents that could invalidate it. If the search results in an infringement and the lawyers have to go to court, the documents must be translated by humans, since MT quality is not good enough for legal purposes.

On the other hand, given the high quality of modern MT systems, MT seems like an obvious intermediate step between being unable to read search results and investing resources in human translation. (In later years, the patent CLIR task was extended to include a PatentMT task [Goto *et al.*, 2011].) In most CLIR research, MT is seen as a separate, post-processing step that is not included in the system evaluation. In this way, researchers can focus on improving intrinsic retrieval without having to take into account the effect of MT errors during result translation.

In this thesis, we argue that result translation is a critical aspect of cross-lingual information access. In addition to focusing on intrinsic measures of retrieval, CLIR and CLQA evaluations should also consider the impact of MT during result translation on the end-to-end system. In these "translingual" evaluations, a result is only useful if it is relevant to the query *and* its translation can be understood by the end user.

In the rest of this section, we discuss traditional CLIA tasks (without result translation). First we provide exact definitions of CLIA, CLQA and CLIR and describe the specific tasks that we use in our experiments. We go into further detail about the GALE distillation tasks because they motivate most of the work in this thesis, and we use queries and corpora from these tasks throughout the thesis. (While our work is strongly motivated by multilingual content on the Internet, our experiments are focused on corpus-based IR, so we do not discuss issues related to web search.) Then we give an overview of CLIR evaluation, including the most commonly used metrics. Finally, we return to the question of result translation for CLIR, which is the focus of this thesis.

2.2.1 Definitions: CLIA, CLQA, and CLIR

We use the term CLIA to cover a range of closely related tasks, including CLIR and CLQA, which are the tasks we focus on in this thesis. Table 2.3 shows the differences between the different tasks we experiment with in this thesis in terms of the following definition.

The goal of a cross-lingual information access system is to satisfy any type of **information need** in the **query language** with **responses** drawn from multilingual corpora, where the corpora are in one or more **document language(s)**, and at least one document language is different from the query language.

An **information need** describes what the user is searching for, and determines which of the responses are correct or relevant. In CLIR, the information need is expressed as a query, whereas in CLQA, it is in the form of a question. In different IR shared tasks, query formats vary quite a bit. For instance, TREC queries contain very short titles, short "descriptions" and long "narratives." On the other hand, user queries tend to be quite short (the average web query is 3 words long), so the information need of the user is not always fully expressed in the query. A user searching for [jaguar] may be interested in comparing prices of cars or writing a report on jaguars in the wild. We discuss the query formats used in our experiments in Section 2.2.2 (and describe them in more detail in individual chapters).

The **query language** and **document language(s)** correspond to the user's language and the corpus language, respectively. These do not exactly match the MT concepts of source and target language, because in an end-to-end cross-lingual application, MT might be used in both directions. When translating the query, the source language is the query language and the target language is the document language; when translating the results back to the user, the reverse is true.

Finally, the **responses** are the results returned by the cross-lingual system from the corpus in response to the information need. A response that satisfies the information need is **relevant** or correct. In many IR tasks, relevance is measured on a scale (for instance, Perfect, Excellent, Good, Fair or Bad), but in this thesis, we only consider binary relevance (each response is either Relevant or Not Relevant). The format of the response is specified by the task. In this thesis, we experiment with both document-level and sentence-level CLIR as well as sentence-level CLQA. We only consider tasks that require retrieving existing items from the corpus. Other cross-lingual tasks outside the scope of this thesis may involve sentence fusion to merge multiple responses, sentence compression

Task	Information need	Response	Relevance
Document-level CLIR	Query	Document	Does the document mention the query?
Sentence-level CLIR	Query	Sentence	Does the sentence mention the query?
Open-ended CLQA	Question	Sentence	Does the sentence answer the question?
Template-based CLQA	Template-based question	Sentence	Template-specific relevance guidelines.

Table 2.3: The cross-lingual tasks used in experiments in this thesis.

to remove redundancy across responses, natural language generation to create a coherent summary out of all of the responses, or sub-sentential extraction to extract exact answers from sentences.

For all of the tasks in Table 2.3, the goal is to find responses in the document language that satisfy the information need in the query language. The table is organized in order of increasing specificity of the information needs. Document-level CLIR merely has to retrieve documents that mention the target, whereas template-based CLQA has very specific relevance guidelines specifying which information should be retrieved.

The challenge in IR and QA is that many relevant responses do not contain the query, and many responses that do contain the query are irrelevant. For instance, a sentence about the "American President" is relevant to a query about Barack Obama, even if the sentence does not mention Obama's name. On the other hand, "Obama made a campaign stop in Columbus, Ohio" is not relevant to the template-based question "List biographical facts about [Barack Obama]." Matching an information need to relevant responses is even harder in the cross-lingual case due to the language gap.

2.2.2 Template-Based CLQA

Much of the work in this thesis was done in the context of the DARPA (Defense Advanced Research Projects Agency) GALE (Global Autonomous Language Exploitation) program, specifically the distillation shared task. These tasks were the first large-scale shared task (that we know of) that specifically focused on returning translated results for CLQA. Previous efforts in evaluating results in translation include the Interactive Track at the CLEF (Cross-Language Evaluation Forum) [Oard and Gonzalo, 2002; Oard and Gonzalo, 2003], but they focused on in-depth user studies, so the evaluations were on a much smaller scale.

The GALE task was template-based CLQA, where queries were significantly different than typical TREC (Text Retrieval Conference), NTCIR (NII Test Collection for IR Systems) or CLEF-like queries. In template-based CLQA, questions are based on pre-defined templates with argument slots, e.g., "Describe the connection between [event/topic X] and [event/topic Y]" (the TREC complex interactive QA (ciQA) task [Kelly and Lin, 2007] used template-based queries of the same style). For each GALE template, there is a specific set of relevance guidelines specifying what kind of information is relevant and what is not. For example, template 7 is "Describe involvement of [person/organization/country] in [event/topic]." An excerpt of the GALE Relevance Guidelines states:

> For a country to be involved in an event/topic, there must exist an official state action regarding the event. The involvement of ordinary citizens (of the country) in the event does not constitute that country's involvement in the event. . . Background information about the event or the involved people, organizations, and countries, is irrelevant if it does not connect explicitly with some involvement in the event.

This is a very narrow definition of relevance, which turned out to be a major challenge in the evaluation. The question templates were motivated by real-world needs of intelligence analysts and evolved over the 5-year span of the GALE program. Appendix A lists the templates for the second and fourth years of GALE (Y2 and Y4), which we use in experiments in later chapters.

2.2.3 IR Metrics

A typical IR evaluation consists of a corpus and a set of queries. All the competing systems return a set of ranked results for each query and the top k results for each query from each IR system are "pooled" for human annotation. Relevance judgments may be binary (relevant/not relevant) or non-binary (like the scale we mentioned above), though in this thesis, we use only binary judgments.

Then for each system run, a variety of metrics can be calculated across annotated results for all the queries in the test set.

The simplest metrics are precision and recall, which measure the fraction of retrieved results that are relevant and the fraction of relevant results that are retrieved, respectively. In IR evaluations, precision is commonly reported out of the top-k ranked results, for instance, as precision at 10.

2.2.3.1 NDCG

Normalized Discounted Cumulative Gain (NDCG) [Järvelin and Kekäläinen, 2000] is an IR metric that takes into account the relative ranking that each system gives to the returned documents. The NDCG at n for query Q is defined by the following formula, where $rel(i)$ is the relevance judgment of the document at rank i, and Z is a normalization factor that makes it so the perfect ranking gets an NDCG score of 1.

$NDCG(Q) = Z \sum_{i=1}^{n} \frac{2^{rel(i)}-1}{log(1+i)}$

If there are 5 relevant items in a ranked list of 10 search results, the precision at 10 is 0.5, regardless of the rank of the items. The NDCG at 10 will be higher for a list where only the top 5 results are relevant than for a list where only the bottom 5 results are relevant. Furthermore, NDCG takes into account the total number of possible relevant results. If there are only 5 relevant results in the entire corpus, a perfect ranking at 10 is one where the first five results are relevant, which would get an NDCG at 10 score of 1.0.

2.2.3.2 MAP

Mean average precision (MAP) is an IR metric that takes into account both recall and precision [Manning *et al.*, 2008]. MAP is a standard metric for evaluating ranked results that is commonly used in IR evaluations. This metric summarizes the overall performance of the system by taking the mean over all queries of the average precision across all levels of recall.

More formally, for a query q, denote by Rel_q^{HT} the set of all HT sentences that are relevant to q. Then for each result d_q in a set of n ranked results, relevance, precision at k and average precision are defined as:

$$rel^{HT}(d_q) = \begin{cases} 1, & \text{if } d_q \in Rel_q^{HT} \\ 0, & \text{otherwise} \end{cases}$$
$$Prec(k) = \frac{\sum_{i=1}^{k} rel^{HT}(d_q^i)}{k}$$
$$AvePrec = \sum_{k=1}^{n} \frac{(Prec(k) \times rel^{HT}(d_q^k))}{|Rel_q^{HT}|}$$

MAP is the average of *AvePrec* over all the queries. The advantage of MAP is that it “provides a single-figure measure of quality across recall levels. Among evaluation measures, MAP has been shown to have especially good discrimination and stability” [Manning *et al.*, 2008].

2.2.4 Result Translation for CLIA

The goal of CLIR and CLQA is cross-lingual matching – the evaluations are done in the source language rather than in translation, so they do not involve task-embedded MT. For the example in Figure 1.1, this would mean just returning the Arabic results. The assumption is that the output of the cross-lingual system can simply be passed to a general-purpose MT system for result translation before being presented to the user.

However, assessing cross-lingual task performance without result translation is problematic, because in a real-world application, result translation does affect task performance. Wang and Oard [2001] showed that users who had access to full translations of CLIR results were able to make better and faster relevance judgments than those just presented with word-for-word glosses.

Part of the reason for the separation between cross-lingual tasks and MT is that evaluating task performance on MT is very difficult. In the 2005 Multilingual Summarization Evaluation (MSE), the task was to produce a multi-document summarization, where the corpus was partly English and partly machine-translated English. Daumé and Marcu [2006] found that doing a manual Pyramid annotation on MT output was very difficult due to the poor MT quality. Furthermore, they found that “it is unclear whether it is necessary or *wise* to use the translated data,” since systems that only selected from English source text performed the best. In a similar task, Evans and McKeown [2005] could not even annotate system output because “machine translations were too difficult for human annotators to understand,” so reference translations were used instead.

While evaluating task-embedded MT has proved difficult, it is not impossible. The interactive cross-lingual task at the Cross-Language Evaluation Forum (iCLEF) studied cross-lingual applications from a user perspective. In the first 5 years (2000 – 2005), iCLEF ran shared tasks involving CLIR and CLQA with result translation, or what they called "task-situated machine translation," with a focus on "user studies for end-to-end cross-language search assistance systems" [Oard and Gonzalo, 2002; Oard and Gonzalo, 2003]. Unfortunately, text-based iCLEF evaluations ended in 2005 because "participation in this track remained low" due to the resources required to build the systems and run the evaluations [Gonzalo *et al.*, 2005].

Although evaluating task-embedded MT can be difficult, such evaluations can provide important information from both the task perspective and the MT perspective. An extrinsic evaluation of an MT system can show how usable the MT output is in a given task context, while evaluating CLIR and CLQA systems with result translation indicates how well these systems would do in a real-world application, where results would have to be translated before being displayed to the user. Bringing CLQA and MT together can also offer opportunities for synergies in the end-to-end system: information from MT can be used to improve result relevance, and information from the CLQA context can be used to improve result MT. In other words, by taking a holistic approach to task-embedded MT, both task performance and MT can be improved.

Chapter 3

The Impact of MT Errors on CLQA

In the previous chapter, we gave an overview of the separate but related fields of CLQA and MT. Task-embedded MT bridges the gap between these two fields by using information from MT to improve task performance and leveraging the task context to improve MT output. In this chapter:

1. We describe the MT systems that will be used throughout the rest of the thesis.
2. We provide further motivation for task-embedded MT by presenting detailed error analyses across three of these MT systems, from the perspective of CLQA with result translation.

Working with different MT systems enabled us to make sure our algorithms were general enough to apply to a variety of MT systems, rather than being system-specific. Furthermore, by doing error analysis on state-of-the-art MT systems over time (in 2006, 2009 and 2011), we ensured that the problems we were working on were not addressed by the MT decoders themselves. In task-embedded MT, application developers typically do not have the opportunity to retrain or re-tune their MT systems; instead, they must use black box (or glass box) MT systems. For that reason, we are not interested in how to build better MT systems, but rather how to adapt the output of existing MT systems to the needs of our task. The error analyses in this chapter explore reasons why the output of task-agnostic MT systems is not always suitable for CLQA with result translation.

In the first error analysis, we show how MT errors during corpus translation can sharply reduce the recall of a CLIR system. In the second analysis, we examine multiple types of MT errors that degrade translation adequacy. Translation adequacy is particularly important for CLQA with result translation, since an inadequate translation may make a relevant sentence appear irrelevant. In the third analysis, we focus on adequacy errors that arise from missing content words. Before describing each MT system's error analysis in detail, we provide an overview summarizing the different MT systems.

3.1 Overview of MT Systems

Table 3.1 lists the various SMT systems we use in experiments in this thesis. All of them were used for Arabic-English MT, and the RWTH system was also used for Chinese-English MT. Glancing at all the MT systems, one might ask, why do we use so many different MT systems? These experiments were carried out over a period of six years (2006-2012), during which SMT systems improved considerably. In order to keep up with the state-of-the-art, updated MT systems were used in each new set of experiments.

The chronological list of MT systems in Table 3.1 follows the development and evolution of different SMT models over the same time period. The RWTH, DTM2 and Moses systems are all non-hierarchical phrase-based SMT systems, while the HiFST system is a hierarchical phrase-based SMT system. The models differ in the algorithms used for training and decoding, as well as the number and types of features used. The systems differ in the amount of training data used; generally, later systems were able to use more training data, as more parallel data became available to the community. Similarly, later MT systems were usually able to harness more computing power, as computers became faster and parallel computing over large clusters became standard.

Running our experiments on such a variety of MT systems encouraged us to keep our algorithms and techniques general enough to work with any SMT system and to handle errors that are common to many SMT systems, rather than addressing system-specific errors that would be solved by switching to a different MT system. For our purposes, the exact implementation of each SMT system does not matter; we wish to understand the impact of MT on CLQA with result translation, and find MT system-independent ways to intelligently combine MT and CLQA to maximize the end-to-end translated relevance.

The only requirement we have for the MT systems is that they produce word- or phrase-level word alignments along with the translated sentences, which almost all SMT systems do. The reason we use alignments produced by MT systems rather than finding aligned phrases using Giza is that the accuracy of our systems is directly related to the accuracy of the alignments. MT system alignments will correspond directly to the translation phrases used to construct the translation, while Giza alignments may have errors. On the other hand, MT system alignments may be more or less informative, depending on their granularity: large phrase-to-phrase alignments are less informative than exact word-to-word or short phrase-to-phrase alignments.

3.1.1 MT System Descriptions

Table 3.1 lists all the MT systems we use in this thesis and which chapters they are used in.

The **RWTH model** is a two-pass, phrase-based model [Mauser *et al.*, 2006; Bender *et al.*, 2007] (named after the Rheinisch-Westfaelische Technische Hochschule university in Aachen). In the first pass, beam search is used to construct an N-best list using six types of feature functions. In the second pass, the N-best list is re-ranked using four types of feature functions. All the feature functions are surface-level only; no part of speech (POS) tags tags or syntax features are used.

The **IBM Direct Translation Model (DTM2)** is a maximum entropy, phrase-based model, which incorporates millions of binary features [Ittycheriah and Roukos, 2007]. The features are based on surface strings as well as bilingual POS tags. The 15 features fall into five categories: lexical features, lexical context features, Arabic segmentation features, POS features and coverage features. The system uses a beam search decoder that has two parameters: the skip length specifies how many source words may be left untranslated, and the window width controls how many source words are considered for translation.

The **Columbia Moses system** is a phrase-based model based on the Moses implementation [Koehn *et al.*, 2007], which has become the standard open source SMT system used by many SMT researchers. Moses uses a stack-based beam search based on several types of feature functions, and the feature weights are tuned for BLEU score using minimum error rate training (MERT). The phrase translation features include a phrase penalty, a phrase translation probability (in both directions, $p(e|f)$ and $p(f|e)$) and a lexical weight (again, in both directions). The other features are: language model, distance-based re-ordering model, word penalty, and lexicalized re-ordering model (which includes six scores).

The **HiFST model** is a hierarchical phrase-based model that is implemented using finite state

MT system	Year Built, Type	Description	Experiments
RWTH	2006, Production	Non-hierarchical two-pass phrase-based MT system. [Mauser *et al.*, 2006; Bender *et al.*, 2007]	Chapter 4 & 5
IBM DTM2	2009, Production	A maximum entropy phrase-based MT system that incorporates millions of source- and target-language binary features. [Ittycheriah and Roukos, 2007]	Chapters 5 & 6
IBM DTM2	2009, Research	Similar to the production DTM2 system, but incorporates additional syntactic features and does a deeper beam search. The research system is also re-trained on training data that has been filtered to match the test data. [Ittycheriah and Roukos, 2007]	Chapters 5 & 6
Columbia Moses	2011, Research	Non-hierarchical phrase-based MT system built using Moses. [Koehn *et al.*, 2007]	Chapters 6 & 7
Cambridge HiFST	2011, Research	Hierarchical phrase-based MT system implemented using FSTs, with two-pass decoding. [de Gispert *et al.*, 2010]	Chapters 6 & 7

Table 3.1: Description of SMT systems used in the error analyses, and subsequently in experiments throughout the rest of the thesis.

transducers (FST) [de Gispert *et al.*, 2010]. HiFST augments the phrase-based translation models with more powerful hierarchical translation rules, which can represent long-distance re-ordering better and may produce more general translation rules than non-hierarchical phrase-based models.

The MT systems are further categorized as either production or research MT systems. For this thesis, we define **research MT systems** as MT systems that produce high quality translations, but incur a significant cost in terms of resources and speed. They are typically built for a shared evaluation, so they can be tuned to a particular genre or even to a specific corpus. In contrast, **production MT systems** produce translations very rapidly and must be able to handle open-domain input. These systems produce translations faster than research systems by reducing the search space or using smaller, lower-fidelity models, for instance a translation model with fewer features or a lower n-gram language model. Research systems are appropriate for MT shared tasks, where a small set of sentences (typically fewer than 1,000) must be translated in a week. Production systems are appropriate for commercial uses, such as web translation, or large-scale corpus translation, as in the GALE distillation task. Typically, production SMT systems are fast enough to do online (real-time) translation, whereas research SMT systems may not be. For the purposes of this thesis, a research MT system is better than a comparable production MT system.

3.2 RWTH Error Analysis

In this error analysis, we first look at the official results of a CLQA shared task. The task is template-based CLQA, which has the most specific information needs of the different cross-lingual tasks we examine (as described in Table 2.3). The poor performance of our CLQA systems on cross-lingual queries compared to monolingual queries motivated an analysis of errors made by our MT system. This second part of our error analysis focuses on named entity (NE) mistranslations.

3.2.1 Template-Based CLQA Task

The Global Autonomous Language Exploitation second year (GALE Y2) distillation task is defined as follows: given a template-based query in English and a specified source corpus, return a ranked set of English snippets that answer the query. The entire distillation corpus contains documents in Arabic, Chinese and English, across two modalities (speech and text) and two genres (formal and informal). The queries are restricted to answers from a specific source section of the corpus (e.g., “Arabic formal text (newswire)” or “any Chinese document”).

Evaluation Corpus: The webtext (informal text) corpus contains approximately 100k web postings in English and Chinese, with roughly 60k web postings in Arabic. The newswire (formal text) corpus contains approximately 100k news stories in each language. The audio corpora contain approximately 40 hours of broadcast conversation (informal speech) in each language and 40 hours of broadcast news (formal speech) in each language.

Evaluation Queries: In the GALE Y2 official evaluation, there were 17 different templates, each of which had specific relevance guidelines. There were 62 queries in the blind test set. The responses were judged by a third party (BAE systems), which also created the queries.

3.2.2 CLQA Approach

The Nightingale team’s approach to the task combined a document translation CLIR system with a variety of template-specific QA systems. First, all the Arabic and Chinese documents were translated to English using the RWTH production system. Then, all the documents were indexed in English using Indri [Strohman *et al.*, 2005]. At query time, the English query was used to retrieve English documents from the Indri index [Kumaran and Allan, 2007], and then template-specific QA systems extracted relevant sentences from the translated English documents.

3.2.3 CLQA Results

Figure 3.1 shows distillation recall and precision of the CLQA system by source language. Queries over Arabic and Chinese documents performed significantly worse than monolingual (English) queries. The low recall was due to the fact that many of the queries returned zero results, including seven Chinese queries, two Arabic queries and one mixed Arabic/English query. Over 40% of Chinese-only queries had zero responses. Other participants in the shared task had similar problems: Boschee *et al.* [2010] describe six queries where none of the participants returned any correct answers.

Even though it was a CLQA task, many of the failures were due to recall errors in CLIR: if the CLIR system did not return relevant documents, the CLQA system had no chance of extracting relevant snippets. Over 90% of the template-based queries had NEs in them, but the CLIR system

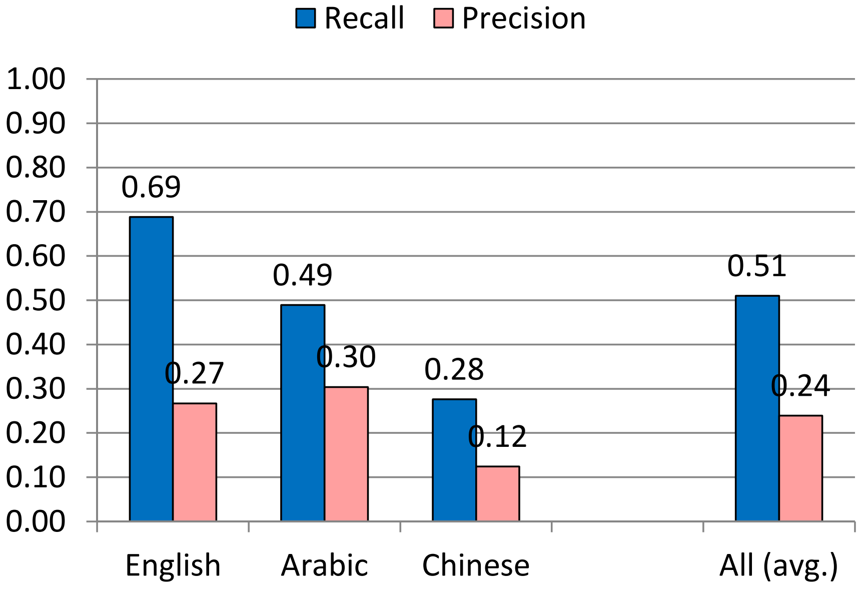


Figure 3.1: GALE Y2 distillation precision and recall, by source language.

frequently failed to retrieve documents containing the NEs in the queries. The document translation approach to CLIR was not robust to errors during corpus translations: if a NE in a document was mistranslated prior to indexing, then at query time, a query containing the NE would not retrieve that document.

In other words, since the index only contained document translations, if the NE was mistranslated during MT, the index actually did not contain the NE at all – no indexed document existed that contained the NE. This shows how MT errors made by a task-agnostic MT system can have a significant impact on task performance: in this case, retrieval recall was seriously degraded by NE mistranslation.

3.2.4 MT Errors: NE Mistranslation

While NEs are crucial to the CLQA task, they are also very difficult to translate correctly, as described extensively in related work [Hermjakob *et al.*, 2008; Habash, 2008]. In this case, the MT system was developed independent of the distillation task, so even though the distillation task required high-quality NE translation, the MT system had no special handling for NEs. NE mistranslations affected all later steps in the CLQA processing pipeline: document-level CLIR, NE recognition, co-reference resolution and sentence-level CLQA.

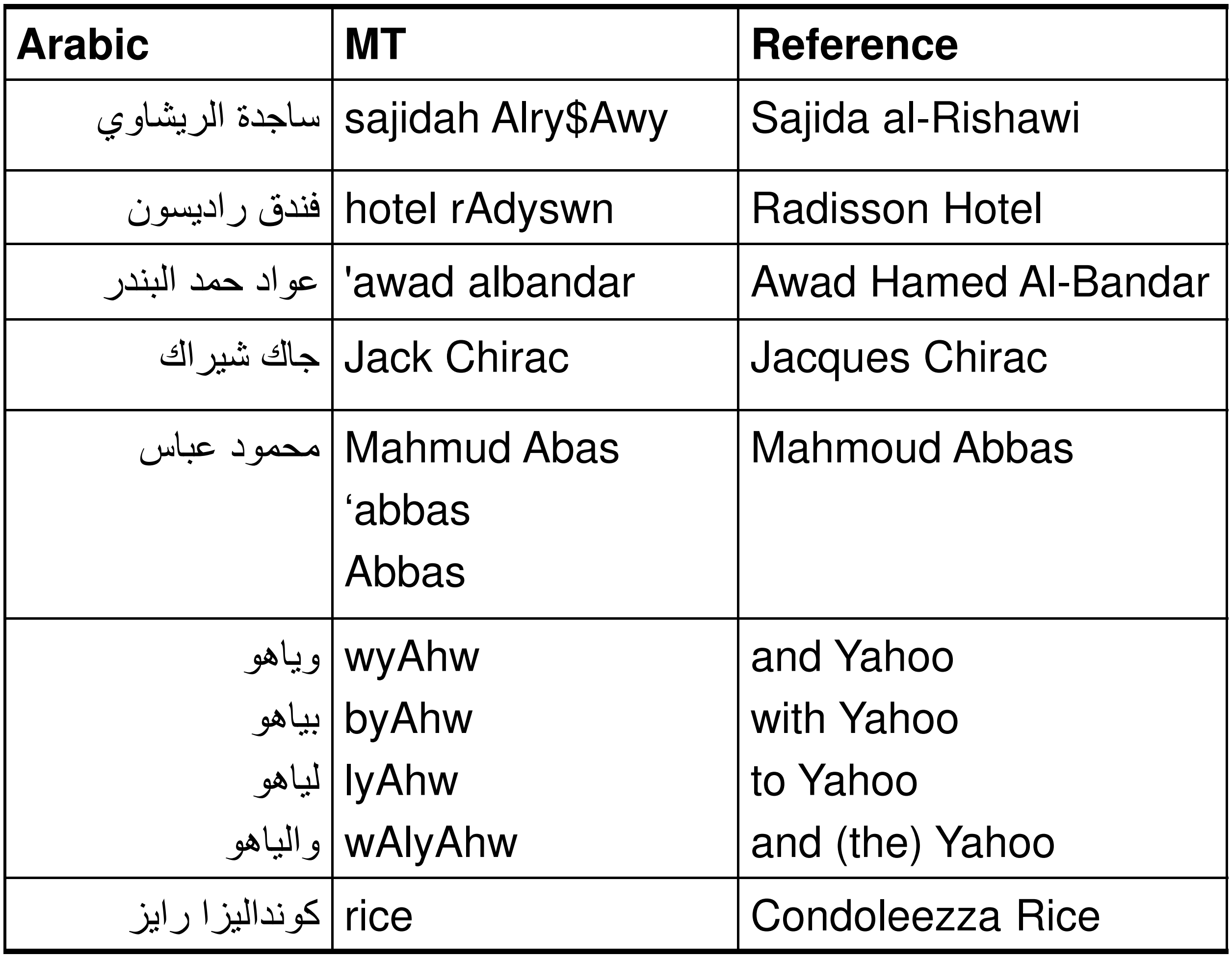
Figure 3.2 shows examples of mistranslated NEs from the distillation corpus. These MT errors are due to a variety of challenges, in particular:

- Many names were **out of vocabulary (OOV)** to the MT system, meaning they were not seen in the MT training data. The RWTH system handled OOV words by transliterating them into Buckwalter [Habash, 2010], which is a case-sensitive mapping from Arabic letter to ASCII letters and punctuation marks. This is problematic for a variety of reasons. First, the transliteration does not correspond to English pronunciation – for instance, Yahoo becomes yAhw and Radisson becomes rAdyswn – so users would not be able to guess the correct name from the Buckwalter. Second, some letters are transliterated to punctuation marks, which is confusing to users and to all downstream processes. The word "Alry$Awy" could be discarded during indexing, since most punctuation is not indexed, which would make it impossible to retrieve with a query. During tokenization, it would likely be split into three separate tokens (Alry, $ and Awy), which would lead to very garbled part of speech tagging and parsing. Third, since Buckwalter is case-sensitive and contains punctuation, NEs that are transliterated using Buckwalter are unlikely to get recognized by the NE recognizer.

- **Arabic morphology** was not handled well by the MT system. This can be seen with the Yahoo example, where the prefixes w+, b+, l+ and wAl+ were not split from the known word Yahoo, and therefore became OOV.

- **NE translation ambiguity** led to mistranslations. In Arabic, Jacques and Jack are spelled the same, which resulted in the surprising mistranslation "Jack Chirac" instead of "Jacques Chirac." Another problem with ambiguity occurs when Arabic names have multiple valid spellings in English. This means that queries searching for Mohamed must also include query terms for Mohammed, Muhamed, etc. Since MT systems translate each sentence separately, the same NE may be translated differently in the same document. Mahmoud Abbas was translated three different ways in a single document, which led to poor co-reference resolution: the entity recognizer identified each misspelling as a different person.

- **Ambiguity between NEs and nouns** also led to mistranslations. Brad Pitt's name was



Arabic	MT	Reference
ساجدة الريشاوي	sajidah Alry$Awy	Sajida al-Rishawi
فندق راديسون	hotel rAdyswn	Radisson Hotel
عواد حمد البندر	'awad albandar	Awad Hamed Al-Bandar
جاك شيراك	Jack Chirac	Jacques Chirac
محمود عباس	Mahmud Abas ‘abbas Abbas	Mahmoud Abbas
وياهو بياهو لياهو والياهو	wyAhw byAhw lyAhw wAlyAhw	and Yahoo with Yahoo to Yahoo and (the) Yahoo
كونداليزا رايز	rice	Condoleezza Rice

Figure 3.2: Examples of named entity mistranslation by the Arabic-English RWTH MT system.

translated as “refrigerator house” because it was handled as two unigram nouns instead of a bigram NE.

- **NE deletion** during translation made relevant documents impossible to retrieve, as there was no indication that the name was even present. NEs could be deleted entirely or partially (“rice” instead of “Condoleeza Rice”).
- **Capitalization** was a problem, because even when an NE was translated correctly, it was frequently not capitalized during re-casing (e.g., albandar), which meant that it was not recognized as an NE. Similarly, inconsistent hyphenation in Arabic names (al-Bandar, albandar, Al Bandar) also caused problems for the NE recognizer.

3.2.5 Summary

The Nightingale team used a document translation approach to do document-level CLIR, followed by monolingual sentence-level QA (in English). While this pipeline worked well for monolingual queries over English documents, queries over Arabic or Chinese documents had much lower recall. Our analysis showed that MT errors led directly to recall failures in the CLIR step: if a NE was mistranslated during document translation, the correct translation of the NE would not be indexed, and therefore could never be retrieved. The MT system and the IR system were developed in a task-agnostic manner and were independent of each other. Even though the task was focused on queries with NEs, the MT system had no special handling for NEs, resulting in the myriad types of NE mistranslations that we describe above. The IR system was unaware of certain aspects of MT output – for instance, the indexer did not recognize Buckwalter-style transliterations and could not index them properly because they contained punctuation. More crucially, the CLIR system never exploited source-language data, such as the Arabic (or Chinese) source documents, so it was completely dependent on the MT system.

3.3 IBM DTM2 SMT System

This error analysis was motivated by the GALE Y4 distillation task, which was also a template-based CLQA task. The IBM DTM2 production SMT system was used by the Rosetta team for the official evaluation. In this analysis, we ignore the impact of CLIR and focus on errors that

High	The error seriously impedes understanding of the sentence; it either changes the meaning or renders it incomprehensible.
Medium	The error makes the sentence hard to understand/parse, but with a bit of (human-level) thinking, one can figure out the meaning/gist.
Low	The error makes the sentence awkward or problematic, but a human can understand the sentence with relative ease.

Table 3.2: Labels for annotating the level of impact of each MT error.

impede understanding of translated sentences, assuming they are retrieved. We expanded the error analysis beyond NEs to include seven types of MT errors that degrade adequacy. The translated data was sampled from separate newswire and webtext corpora, and we present the results from each genre separately before discussing the overall results.

3.3.1 Data

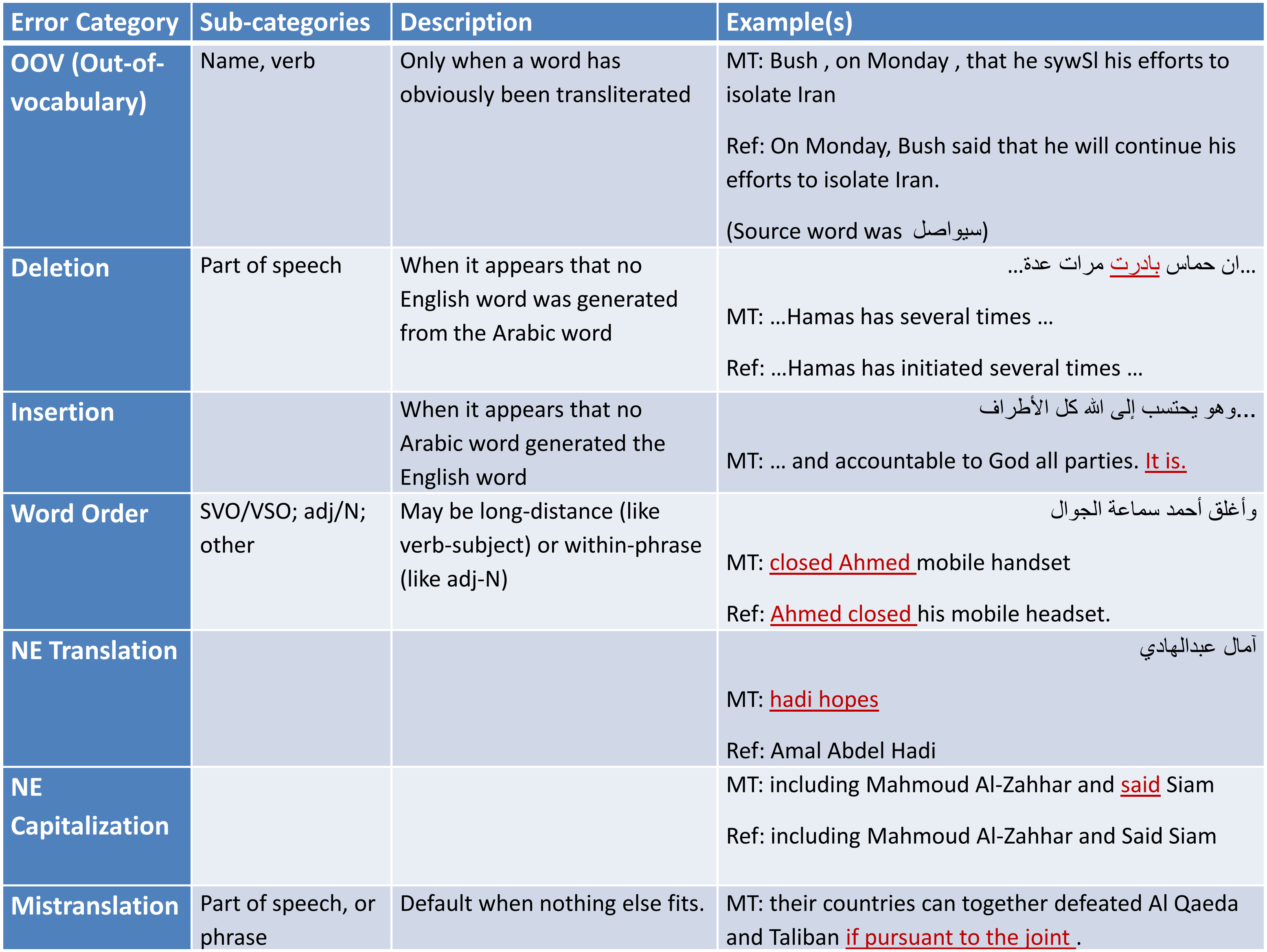
The newswire translations were extracted using two CLQA queries over the Arabic newswire GALE Y4 corpus. For each query, the system returned response sentences as well as additional supporting sentences and adjacent sentences, for a total of 62 sentences overall. Since the sentences are all related to the two queries, many of them are near-duplicates or mention similar topics.

Since no webtext queries were available at the time, the webtext translations were sampled from an MT development test set. Every twentieth sentence from the NIST MT 2008 webtext evaluation set was extracted, for a total of 27 sentences. Unlike the newswire sample, these sentences are all from different documents, so there is no repetition or similar topics.

3.3.2 Error Annotation

The goal of the annotation was to quantify and categorize MT adequacy errors; in other words, those that caused problems in comprehension. Minor grammatical errors were not annotated. Each error was categorized as one of the seven error types in Figure 3.3, four of which had additional sub-categories. Each error was also annotated with an impact level, described in Table 3.2.

For the newswire corpus, MT sentences were annotated based only on the MT and the Arabic



Error Category	Sub-categories	Description	Example(s)
OOV (Out-of-vocabulary)	Name, verb	Only when a word has obviously been transliterated	MT: Bush , on Monday , that he sywSl his efforts to isolate Iran Ref: On Monday, Bush said that he will continue his efforts to isolate Iran. (Source word was سيواصل)
Deletion	Part of speech	When it appears that no English word was generated from the Arabic word	...ان حماس بادرت مرات عدة... MT: ...Hamas has several times ... Ref: ...Hamas has initiated several times ...
Insertion		When it appears that no Arabic word generated the English word	...وهو يحتسب إلى الله كل الأطراف MT: ... and accountable to God all parties. It is.
Word Order	SVO/VSO; adj/N; other	May be long-distance (like verb-subject) or within-phrase (like adj-N)	وأغلق أحمد سماعة الجوال MT: closed Ahmed mobile handset Ref: Ahmed closed his mobile headset.
NE Translation			آمال عبدالهادي MT: hadi hopes Ref: Amal Abdel Hadi
NE Capitalization			MT: including Mahmoud Al-Zahhar and said Siam Ref: including Mahmoud Al-Zahhar and Said Siam
Mistranslation	Part of speech, or phrase	Default when nothing else fits.	MT: their countries can together defeated Al Qaeda and Taliban if pursuant to the joint .

Figure 3.3: Categories for annotating MT errors for the IBM DTM2 error analysis.

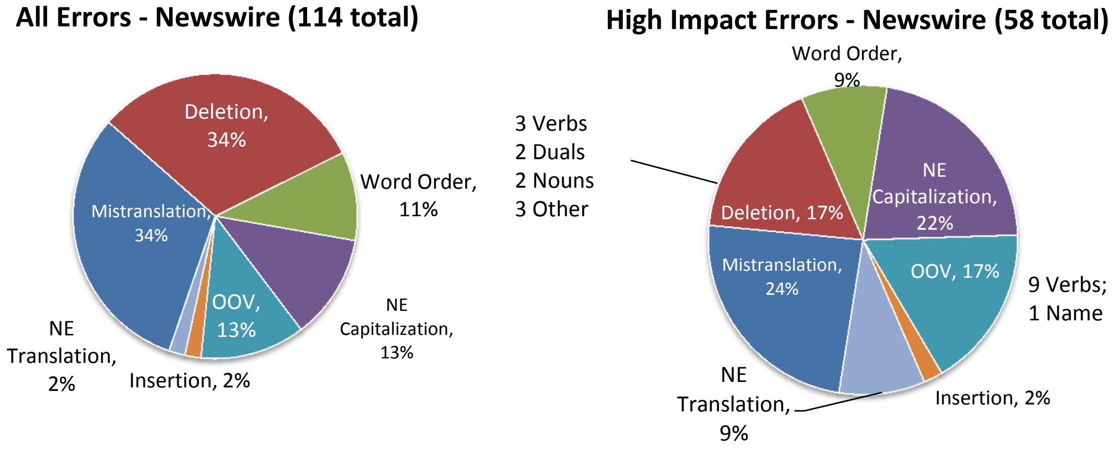


Figure 3.4: Manually annotated MT errors for IBM DTM2 newswire sample. (Table 3.2 defines what a high impact error is.)

source, while for the webtext corpus, four reference translations were also available. Unfortunately, word alignments were not available for these corpora, so the error analysis was descriptive rather than diagnostic.

3.3.3 Error Analysis Results: Newswire

Figure 3.4 shows the frequency of each type of MT error in the newswire sample for all errors and for high impact errors. Overall, 87% (54 / 62) of newswire sentences were annotated with one or more MT errors. About half of them were high impact errors (58 / 114 or 51%) (defined in Table 3.2).

Almost half of all sentences had at least one “mistranslation” error (45%). Of the high-impact mistranslations, many were lexical (36%), meaning that the system chose the wrong word or bigram in translation. There were also many phrase-level mistranslations (29%), which indicated an error in translating “non-dictionary” multiword expressions. (For example, “pursuant to the joint” instead of “if they work together.”)

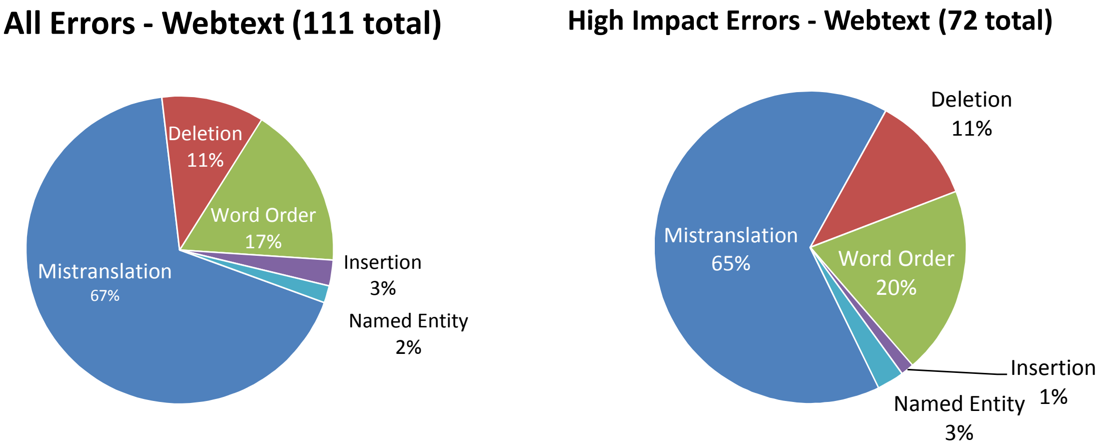
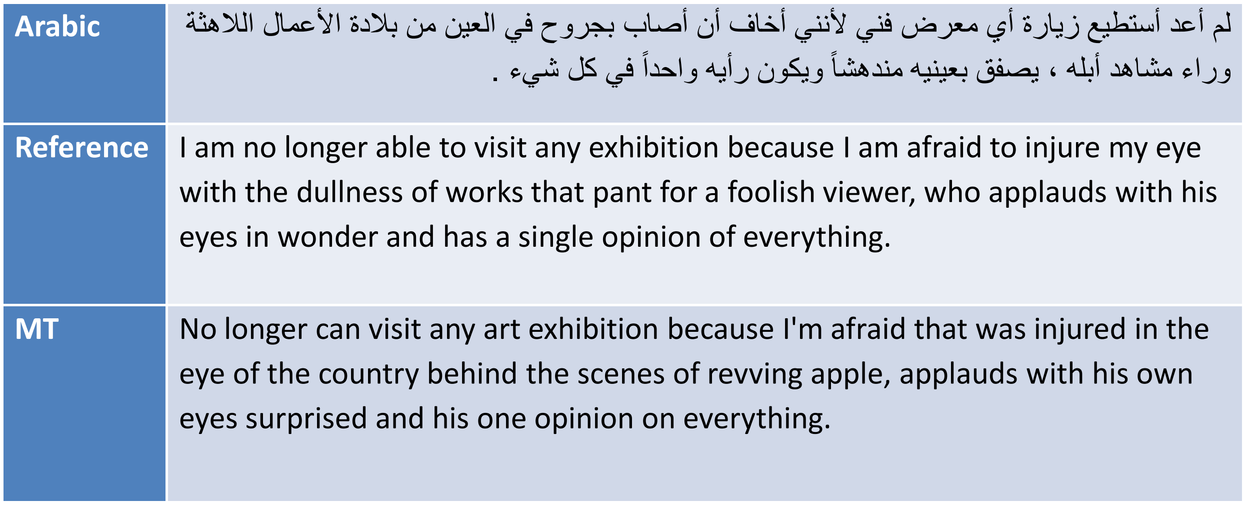


Figure 3.5: Manually annotated MT errors for IBM DTM2 webtext sample.

A third of all sentences had at least one deletion error (35%), and deletions accounted for 17% of the high-impact errors. The main problems were content word deletions (verbs, nouns, and phrases), but deleting smaller words/tokens could also cause confusion. For example, dropping the dual ending caused "two days" to get translated as "one day" in two separate examples.

Garbled word order was a common problem, affecting 23% of all sentences, but was not as significant as a high-impact error. The most common word-order problem was VSO sentences in Arabic that were not reordered to SVO in English. Often, it seemed that the reader could "figure out" or undo the mix-up, so these errors were not as crucial for understanding, though they would be essential for grammaticality (and BLEU score). Awkward placement of prepositional phrases (for example, between a subject and verb or a verb and object) was also an issue.

NE translation was surprisingly good, with only five mistranslated NEs and only one OOV NE. The mistranslated names were mostly due to noun/NE ambiguity (e.g., the name al-Harb was translated to the word "war" because it means "the war"). On the other hand, NE capitalization was very poor, with 13 names presented in lower-case. Although humans would often have no trouble with this, English NE taggers require case information, so it is crucial for the CLQA task to have properly capitalized names. In many cases, a NE would be correctly capitalized in several



Arabic	لم أعد أستطيع زيارة أي معرض فني لأنني أخاف أن أصاب بجروح في العين من بلادة الأعمال اللاهثة وراء مشاهد أبله ، يصفق بعينيه مندهشاً ويكون رأيه واحداً في كل شيء .
Reference	I am no longer able to visit any exhibition because I am afraid to injure my eye with the dullness of works that pant for a foolish viewer, who applauds with his eyes in wonder and has a single opinion of everything.
MT	No longer can visit any art exhibition because I'm afraid that was injured in the eye of the country behind the scenes of revving apple, applauds with his own eyes surprised and his one opinion on everything.

Figure 3.6: A sentence from the MT08 web evaluation corpus. Even in the reference translation, it is difficult to understand.

places, and then lower-cased in others, so this is also a consistency/coherence problem.

OOV words affected 15% of sentences. Nine out of the ten errors were due to verbs which could not be translated, often due to morphology (e.g., future tense, dual ending, etc.).

Finally, a few sentences had insertion errors, where tokens seemed to appear out of nowhere. The five inserted tokens were: right parenthesis (where there were no parentheses, right or left, in the source); m+ (probably an artifact of morphological analysis); comma; "being"; and "said". All except "being" were not high-impact errors.

3.3.4 Error Analysis Results: Web Text

Overall, 85% (23/27) of the sentences were annotated with one or more MT errors. This was highly correlated with sentence length: the average length of no-error sentences was 5 tokens, as compared to 26 for all sentences. However, even some of the shortest sentences had errors. Many of the webtext sentences were so mangled that it was difficult to find specific reasons for the errors. Even with the reference translations, many sentences barely made sense in English, and the idiomatic nature of the genre was much more difficult to understand, as can be seen in the example in Figure 3.6.

Mistranslation was responsible for a large majority of errors – 74% of sentences had at least one mistranslation error, and 59% of sentences had at least one high-impact mistranslation. Poor lexical choice for nouns and verbs accounted for many high-impact mistranslations (38% were nouns, 23% were verbs). Phrases or multiword expressions were another 23% of high-impact mistranslations, with the remaining 16% miscellaneous types of words (prepositions, pronouns, adjectives, and demonstratives).

Word order was more of a problem for webtext than newswire. Most of the high-impact word order problems were due to VSO/SVO not getting reordered (64%), while others were due to misordered noun-adjectives, the idafa construction (a special noun-noun genitive construction in Arabic) or other syntactic structures.

Deletion was again a problem, although the types of words that were deleted were different than newswire. Phrases and multi-word expressions accounted for half of high-impact deletions. Nouns, possessives and pronouns were the remainder. Surprisingly, no verb deletions were noted (although some of them may have been annotated as mistranslations if the whole phrase was garbled).

There were only three insertions (where a word appeared with no obvious source word), but they all mangled the translations significantly. In one sentence, a new sentence "It is." appeared after the translated sentence.

There were only two NE mistranslations, both of which were rare place names. Unfortunately, the system chose to translate the NEs as unrelated words rather than leave them transliterated. (Tahala became "now" and Guelma became "set out.")

There were no OOV errors and no NE capitalization errors in the webtext (partly due to the fact that there were very few NEs in the sample).

3.3.5 Summary

We manually annotated two corpora translated by the IBM DTM2 system for adequacy errors, and then categorized the errors by type and severity. Overall, both corpora suffered from adequacy errors – 87% of newswire sentences and 85% of webtext sentences had at least one error. The webtext was a much more challenging genre for the MT system: the average number of errors per sentence was 4.1 for webtext and 1.9 for newswire (and the average number of high-impact errors was 2.7 and 0.9, respectively). While the newswire sentences tend to be grammatical and well-written, the webtext genre varies widely from well-written, grammatical text to poorly written rants about a variety of topics.

The types of errors also varied significantly between the two genres. Newswire contained many more NEs than webtext, and consequently had many more NE translation and capitalization errors. Newswire suffered from many OOV errors, all of them high impact. Webtext had no recognizable OOV errors, although some of the mistranslation errors could have been due to OOV. On the other hand, garbled word order errors were more prevalent in webtext than newswire. Mistranslation and deletion were major problems for both genres.

3.4 HiFST MT Analysis

In our third error analysis, we focus on MT errors that directly lead to a drop in adequacy: missing and deleted content words. The analysis was carried out on sentences translated with the baseline HiFST MT system as well as with a version of the system that was explicitly tuned to avoid dropped words. In addition to annotating and categorizing MT errors on a small sample of translated sentences, we also looked at automatic metrics and crowd-sourced manual judgments of translation adequacy.

3.4.1 Data

The full corpus consists of newswire articles sampled from the NIST MT 2002-2005 evaluation test sets, for a total of 2,075 sentences. Automatic metrics were calculated over the full set, while the error annotation and crowd-sourced adequacy judgments were carried out over samples of the corpus.

For the detailed manual analysis, 3 chunks of 10 nearby sentences were randomly selected: 50-59, 150-161 (skipping 157 and 160) and 1200-1209. We selected neighboring sentences because they were typically from the same articles and about the same topic, which is similar to the set of related sentences we would expect from CLQA results. Each sentence was annotated based on the Arabic source, the English MT, MT phrase alignments and four reference translations.

3.4.2 Error Annotation

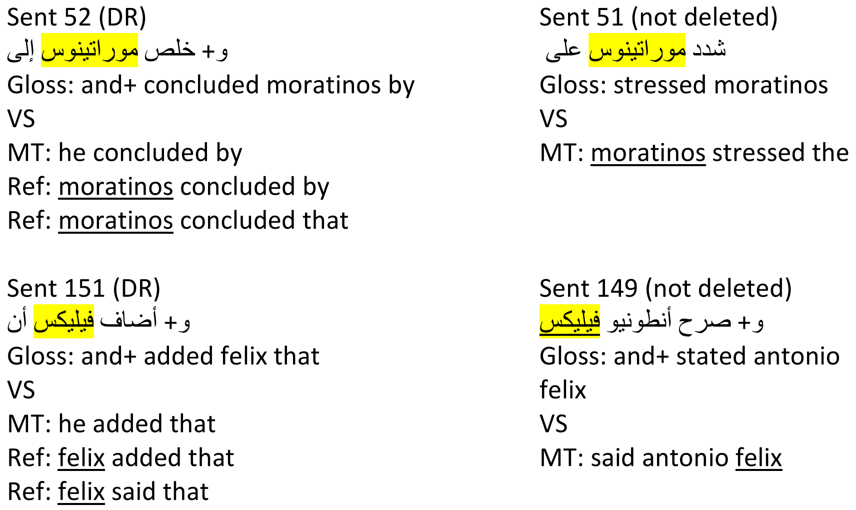
Translated sentences were annotated for three broad categories of adequacy errors: missing content words, mistranslations and garbled word order. Since the goal of this analysis was diagnostic as well as descriptive, we further categorized the missing content words into three categories: intentionally dropped words, OOV words, and deletions due to mistranslations. A "content word" is an Arabic noun, verb or adjective; this does not include function words or pronouns. Each deletion was further labeled as valid or harmful, depending on whether the deleted word was essential for a fully adequate translation.

The "dropped" category is specific to the HiFST system, which has a special mechanism for intentionally dropping words and phrases called a drop rule. The decoder can apply the drop rule to any word or phrase in a sentence, with the constraint that no more than two phrases in a row can be dropped. Since HiFST is a hierarchical phrase-based SMT system, all phrase translations are referred to as "rules" and deletions due to OOV are called OOV rules.

3.4.3 Results: Baseline HiFST

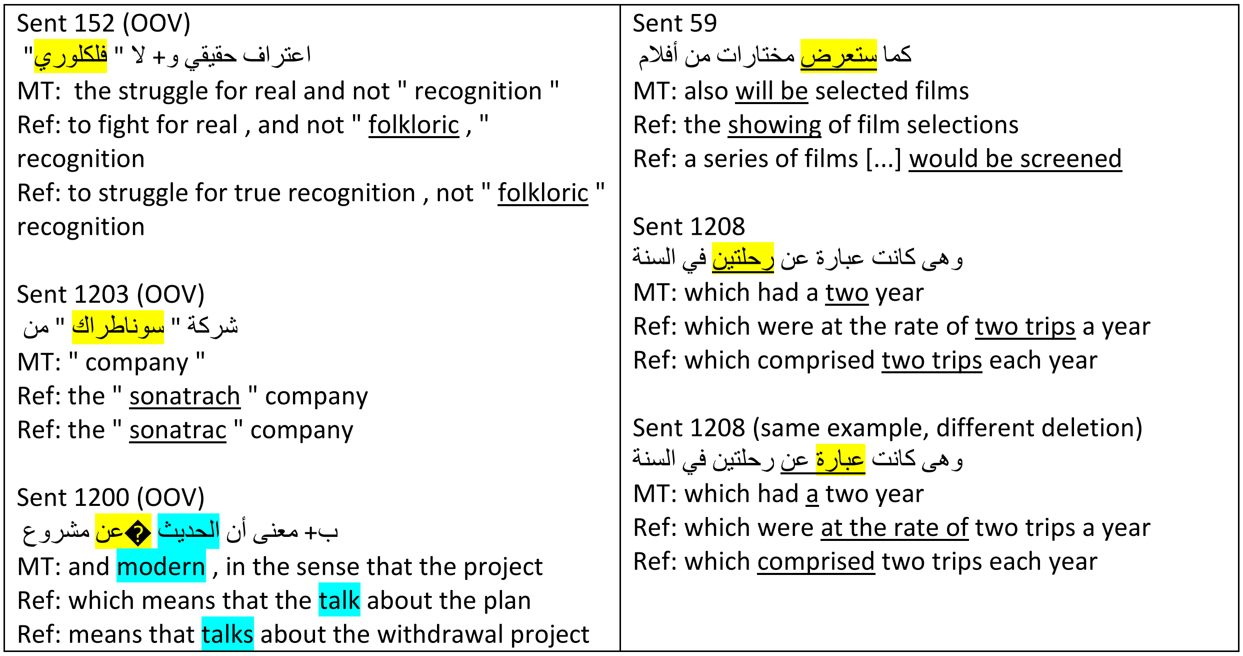
First, we automatically analyzed the full corpus, to estimate how many sentences were affected by OOV and drop rules. In the full corpus, 44% (897 / 2,061) of the sentences had at least one DR or OOV rule applied, which means that these types of deletions have a large impact on translation. (Since these statistics are collected automatically, they do not indicate whether the deletions were harmful to adequacy or not.) Not surprisingly, longer sentences tended to have more deletions: the average length of a sentence with no deletions was 26 Arabic tokens, versus 33 Arabic tokens in sentences with at least one deletion. The majority of these deletions were drop rules (69%) rather than OOV rules (31%), meaning that the often system has some knowledge about how to translate the dropped token, but chooses not to. About 6% of all numerals in the dataset were dropped or OOV (67 / 1,117), even though we expect that numerals should almost never be deleted.

The manually annotated corpus contained 30 sentences. Overall, 40% of the sentences had no major adequacy errors. On the other hand, over half of sentences (53%) had one or more content word deletions that affected adequacy. (The remaining sentences had adequacy errors that were not due to deletions.) Of the content word deletions, the system explicitly deleted the word 75% of the time, either via a drop rule (52% of all deletions) or an OOV rule (23%). The remaining



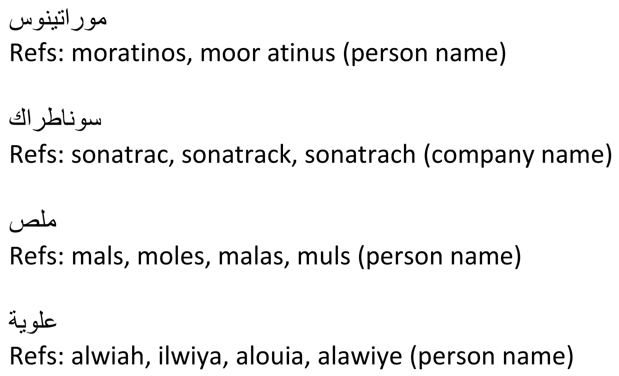
Sent 52 (DR) و+ خلص موراتينوس إلى Gloss: and+ concluded moratinos by VS MT: he concluded by Ref: moratinos concluded by Ref: moratinos concluded that	Sent 51 (not deleted) شدد موراتينوس على Gloss: stressed moratinos VS MT: moratinos stressed the
Sent 151 (DR) و+ أضاف فيليكس أن Gloss: and+ added felix that VS MT: he added that Ref: felix added that Ref: felix said that	Sent 149 (not deleted) و+ صرح أنطونيو فيليكس Gloss: and+ stated antonio felix VS MT: said antonio felix

Figure 3.7: Examples of VSO sentences where HiFST did (or did not) drop the subject.



Sent 152 (OOV) اعتراف حقيقي و+ لا " فلكلوري" MT: the struggle for real and not " recognition " Ref: to fight for real , and not " folkloric , " recognition Ref: to struggle for true recognition , not " folkloric " recognition Sent 1203 (OOV) شركة " سوناطراك " من MT: " company " Ref: the " sonatrach " company Ref: the " sonatrac " company Sent 1200 (OOV) ب+ معنى أن الحديث عن� مشروع MT: and modern , in the sense that the project Ref: which means that the talk about the plan Ref: means that talks about the withdrawal project	Sent 59 كما ستعرض مختارات من أفلام MT: also will be selected films Ref: the showing of film selections Ref: a series of films [...] would be screened Sent 1208 و هى كانت عبارة عن رحلتين في السنة MT: which had a two year Ref: which were at the rate of two trips a year Ref: which comprised two trips each year Sent 1208 (same example, different deletion) و هى كانت عبارة عن رحلتين في السنة MT: which had a two year Ref: which were at the rate of two trips a year Ref: which comprised two trips each year

Figure 3.8: Examples where content words were deleted due to OOV errors and mistranslation errors. The examples on the left show cases where the deletions also affected the translations of surrounding tokens.



موراتينوس
Refs: moratinos, moor atinus (person name)

سوناطراك
Refs: sonatrac, sonatrack, sonatrach (company name)

ملص
Refs: mals, moles, malas, muls (person name)

علوية
Refs: alwiah, ilwiya, alouia, alawiye (person name)

Figure 3.9: Examples from the evaluation corpus where the NE translation was inconsistent among the human translators.

deletions were due to phrase-to-phrase translations that were missing a word on the target side: of 32 overall deletions, all but four of them were harmful.

Drop rule deletions: Most of the words deleted by drop rules were nouns (10/16) or NEs (8/16). In many SMT systems, a word that is deleted in one sentence may be translated correctly in another sentence: of the 15 "harmful" dropped words, at least two of them were correctly translated in other sentences in this same dataset, sometimes in neighboring sentences, as can be seen in Figure 3.7.

OOV deletions: Over half of the OOV words (4/7) were NEs. Translating the OOV names correctly would require transliterating them. In two of these cases and three of the dropped NEs, the reference translators disagreed as to how to spell the name, as shown in Figure 3.9, meaning that there was no standard spelling. Two other OOV words were adjectives (possibly due to morphological endings). The final OOV word was not a content word, but was OOV due to an encoding issue. (At least 20 tokens in the tuning set contained an invalid Unicode character, and all of them were OOV.)

Mistranslation deletions: The final category of deletions consists of deletions via regular translation rules (not drop or OOV rules). Out of 9 deletions, 7 of them were harmful: 5 nouns, 1 verb and 1 verbal noun. In sentence 59 in Figure 3.8, the verb "will be screened" is just translated as "will be." In sentence 1208, the dual noun "two trips" is translated as just "two". In both of these cases, the meaning of the morphological affixes is translated, but the meaning of the stems is lost. In the third example (also from sentence 1208), a noun phrase is translated to a determiner. When a phrase with a content word is translated into a phrase without a content word, it is a strong signal that the content word has been deleted. Other errors were due to phrase-to-phrase translations where only part of the meaning was expressed.

In addition to affecting the adequacy of a sentence, a deletion may also negatively impact the translations of neighboring words. In the first two examples in Figure 3.8, a deletion within quotes caused the quotes to jump to a nearby word. In the third example, the deletion caused the previous word to get mistranslated as “modern”. The deleted word “about” is a strong indication that the previous word should be “talk” — in five other sentences in this dataset, the phrase “talk about” is translated correctly. However, in the absence of the next word “about”, the most common translation for the word is “modern”.

3.4.4 Results: Without Drop Rules

Since drop rules were responsible for the majority of deleted content words in HiFST output, we did a contrastive analysis of the same data translated without drop rules. The drop rule probability was set to zero and the HiFST decoder was re-tuned for BLEU score using MERT, so some other parameters in the system were also changed. We refer to the HiFST system without drop rules as noDR.

Over 58% of the sentences in the full corpus differ from the baseline MT system to the noDR MT system, including sentences that were unaffected by DR rules, since the system was re-tuned. The percentage of sentences where tokens were deleted (either due to drop rules or OOV) decreased from 43% to 17%. Although 1,069 drop rules were no longer used, the overall average sentence length increased only slightly.

Automatic metrics: Table 3.3 compares the automatic metrics scores of the HiFST baseline and noDR systems. The baseline system has a significantly better BLEU score than the noDR system (p=0.05). However, the noDR version does slightly better on METEOR, both on the adequacy-tuned version and the HTER-tuned version, as well as on versions of TERp tuned for adequacy and fluency. (These changes are not significant.)

In order to focus on the impact of the drop rules only, the sentences that originally had drop rules were extracted. There were 736 sentences, or 35% of the corpus. The automatic metric

	sent length	BLEU	TERp-fluency	TERp-adeq	METEOR-HTER	METEOR-adeq	
baseline (full)	33.05	0.5351*	0.3114	0.3426	0.7267	0.6943	
noDR (full)	33.15	0.5311	0.3083	0.3398	0.7273	0.6963	
baseline (sample)	37.34	0.4835*	0.3796	0.4134	0.6785	0.6430	
no-DR (sample)	38.42	0.4788	0.3665	0.4018	0.6818	0.6509	

Table 3.3: Automatic metrics for the HiFST baseline and noDR systems.

scores for this sample are shown in the bottom two lines of table 3.3. After turning off drop rules, the average sentence length increased by more than one token. The BLEU (bilingual evaluation understudy) score is significantly better with drop rules on (p=0.05), but by every other metric, the noDR version does better. The differences on this smaller set are significant but on the full corpus they are not.

While BLEU is based only on precision, METEOR and TERp take both precision and recall into account. The fact that the metrics which include recall improve suggests that the "undropped" translations are often good.

Manual annotation: The 30 sentences that were initially analyzed for adequacy errors were again re-examined, this time comparing the noDR MT with the baseline MT. Each annotated error was marked as *correct* or *not correct*, and each whole sentence was marked as to whether the adequacy was (*better*, *same*, *worse*, or *can't tell*). Even sentences that changed may have the same level of adequacy if the change did not have an impact on adequacy, or they may be mixed if some changes were positive and some were negative.

In the small corpus, 11 out of 30 sentences had drop rules. On the whole sentence level, 5 sentences had improved adequacy, 2 stayed the same, 1 got worse and 3 were mixed or uncertain. Since this was a small sample with one annotator, we decided to collect judgments over a slightly larger corpus with more annotators.

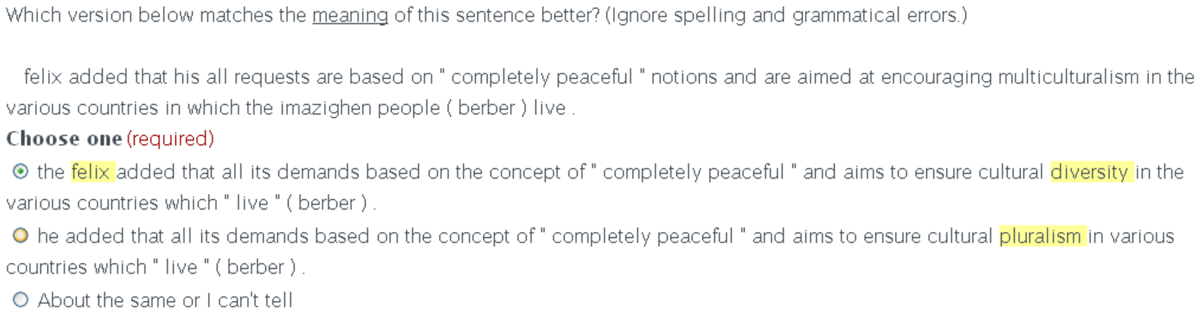
An “undropped” token is one which was dropped by the baseline system but translated in the noDR system. As figure 3.11 shows, out of 16 undropped tokens, 11 of the translations in the noDR system were correct. The remaining 5 were incorrect or dubious translations. Half of the baseline dropped words were NEs, and 6 / 8 of these were translated correctly in the noDR system.

This suggests that the system often has a good translation for the dropped word, but chooses not to use it. To understand why, we can look back at the examples in Figure 3.7. The NEs translated correctly in the right-hand examples are dropped in the left-hand examples (“moratinos” and “felix”). While the actual verb subjects are dropped, the verbs in both cases are translated into bigrams with a spurious subject “he” (“he concluded” and “he added”). To get the translations completely right, the NE must be translated correctly, the verb must be translated without the spurious subject, and the subject and verb must be re-ordered, which only happens in the example from sentence 51.

When translating with the noDR system, we are forcing the MT system to translate the NEs instead of dropping them, which in turn forces the verbs to get translated without the spurious subject. But when translating with the baseline system, the decoder prefers to output “he added” over “felix added” even though it has to drop a word. Any language model would score “he added” much higher than “felix added”, and in general, ”he” will always have higher score than a given NE. In these examples, the drop rules encourage the decoder to delete NEs.

NEs are very difficult for MT systems to get correct, yet they are crucial for sentence understanding, especially in task-based MT. On the other hand, since NEs make up such a small percentage of the tokens in a corpus, they do not significantly impact BLEU score [Papineni *et al.*, 2002a]. Indeed, since many NEs have multiple accepted spellings, BLEU score may penalize NEs even when they are correct, if the spelling does not match the reference [Hermjakob *et al.*, 2008]. In other words, NEs are very risky translations, so SMT systems often delete them, which leads to adequacy errors.

Crowd-sourced judgments: We selected sentences affected by drop rules with length less than or equal to 15 tokens (in the source), yielding a corpus of 77 sentences. We then asked five turkers to select the “best” translation (or mark them the same) based only on meaning, as shown in the annotation interface in Figure 3.10. The order of sentences was randomized, and the



Which version below matches the meaning of this sentence better? (Ignore spelling and grammatical errors.)

felix added that his all requests are based on " completely peaceful " notions and are aimed at encouraging multiculturalism in the various countries in which the imazighen people (berber) live .

Choose one (required)

◉ the felix added that all its demands based on the concept of " completely peaceful " and aims to ensure cultural diversity in the various countries which " live " (berber) .

○ he added that all its demands based on the concept of " completely peaceful " and aims to ensure cultural pluralism in various countries which " live " (berber) .

○ About the same or I can't tell

Figure 3.10: The CrowdFlower annotation interface for contrastive adequacy judgments between the baseline and noDR HiFST MT output.

experiment was run through CrowdFlower, so only "trusted" turkers' judgments were kept. We selected short sentences because they are easier for turkers to annotate (they tend to have higher agreement than long sentences), and they often had fewer drop rules per sentence. As shown in Figure 3.11, noDR was preferred almost half of the time, baseline was preferred nearly one-third of the time, and the sentences were the same about one-fifth of the time.

3.4.5 Summary

The drop rule in HiFST allows the decoder to avoid being forced to use bad translations by choosing to drop words, even when it has some possible translation rules for them. In general, this allows the system to avoid risky translations, or translations that have low LM scores. Tuning the system to a precision-based metric (BLEU) results in a somewhat conservative system that is more inclined to drop "risky" translations, even when they are correct. However, some of the "risky" translations include NEs, which are often crucial to sentence adequacy. The analysis showed that the system often has a good translation for a dropped word (11 / 16 cases). Sentences without drop rules (translated by noDR) are often preferred to the baseline system (46% of the crowd-sourced annotation, 45% of the manual annotation). But when the system does not have a good translation, the baseline MT system (with drop rules) is preferred (31% of the turk annotation).

Turning drop rules off entirely has some negative effects as well. While the METEOR and TERp scores of the sentences affected by drop rules increased significantly, the BLEU score decreased

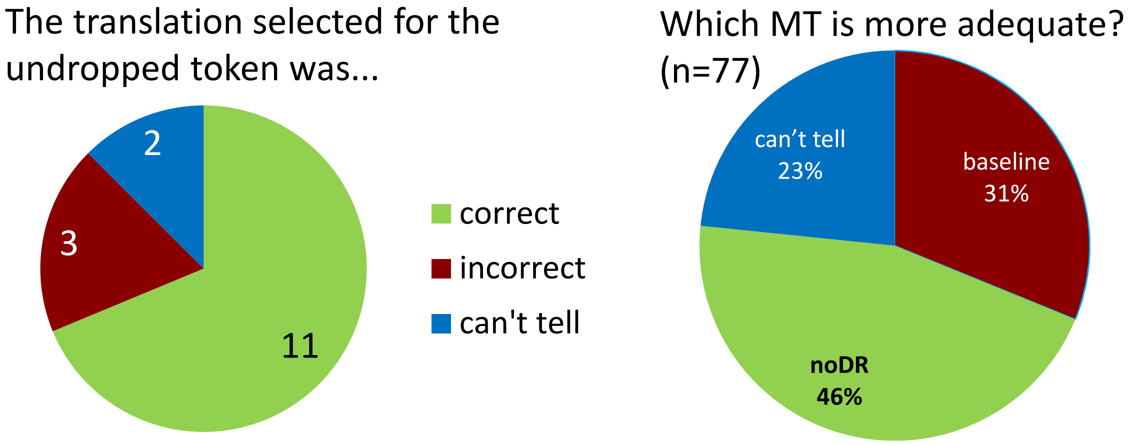


Figure 3.11: The left-hand chart shows manual annotation of the 16 tokens that were dropped by the HiFST system and no dropped by the noDR system. The right-hand chart shows crowd-sourced contrastive adequacy judgments comparing the baseline and noDR output on 77 sentences (of length 15 tokens or less).

significantly. In other words, translation adequacy (recall) is improved at the expense of precision.

Furthermore, the noDR system does not address all content word deletions. Many deletions were due to OOV rules and mistranslations such as translating a content word to a function word or translating a phrase into only part of the phrase. All of these errors degrade adequacy significantly, and present challenges for CLQA with result translation.

3.5 Conclusions

We presented three different kinds of error analyses from corpora translated by three different SMT systems. The analyses described which MT errors were most prevalent, how the MT errors arose, and how they impacted CLQA performance and/or translation adequacy, all of which motivated the experiments in the rest of this thesis. The conclusions we drew from the analyses are as follows:

- **MT errors can significantly degrade CLIR recall.** The RWTH error analysis showed that CLQA performance was much lower on queries over translated documents than on monolingual queries due to a drop in CLIR recall. The document translation approach to CLIR was particularly vulnerable to MT errors, since no source-language data was used. This result directly motivated the SMLIR model that we describe in Chapter 4.

- **Adequacy errors are prevalent in MT output.** All of the MT systems produced sentences with adequacy errors, even as the MT systems improved using better algorithms, more data and faster machines. Lexical mistranslation, content word deletion and garbled word order were the major types of adequacy errors. Our experiments with automatic post-editing (in Chapter 6) were motivated by this observation.

- **NE translation is crucial for CLQA.** Although NEs make up only a tiny percentage of tokens that need to be translated, translating them correctly is essential for NE-oriented CLQA. Translating NEs presents special challenges for a variety of reasons, yet most task-agnostic MT systems do not have special handling for NEs. The query-specific NE post-editor (in Section 6.3) was developed to address this problem.

- **Content word deletion is a common problem, and significantly degrades MT adequacy.** Content words can be deleted several different ways, but they are nearly always harmful rather than valid deletions. Even developing an MT system that tries to avoid dropped words (HiFST noDR) does not solve content word deletion errors. All three SMT systems had NE deletion errors, which is particularly detrimental to the CLQA task. We developed algorithms to detect (Chapter 5) and correct (Chapter 6) content word deletion errors.

Chapter 4

Using MT to Improve CLIR Relevance

In this chapter, we study CLIR in isolation from the rest of the CLQA system in order to focus on improving result relevance independent of result translation. The model we present, SMLIR, was motivated by CLQA with result translation and takes particular advantage of the translated data that is required for this task, which would normally not be available for a traditional CLIR/CLQA task without result translation. In future chapters, we will show how our proposed CLIR model is particularly suitable for the task of CLQA with result translation because it enables us to detect and correct errors in document translations.

In a typical CLQA system architecture, the first step would be to use CLIR to quickly retrieve relevant documents. Then each of the top n documents would be processed by response generators to select the most relevant sentences. The response generators are generally more sophisticated and resource-intensive than IR, so the first-pass CLIR step is necessary to filter the number of documents that need to be processed. In this context, CLIR recall is more important than precision: if an irrelevant document is returned, the response generator can simply not select any sentences from it, but if a document containing a relevant answer is missed by the CLIR system, the sentence selection algorithm will never see it. The first error analysis in the previous chapter (in Section 3.2) indicated that many of the CLIR recall errors were due to NE mistranslation by the MT system. This highlighted problems with a pure document translation approach to CLIR: once a word is mistranslated by MT, there no way to retrieve the document, since all source language information is discarded. This motivated the development of the SMLIR model.

Although CLIR evaluations are typically done in the document language, our proposed model is based on the premise that the ultimate goal of the system is to return translated results. In the absence of an online translation system, the corpus must be machine translated ahead of time to facilitate returning translated documents, so we assume that a full corpus translation is available. Our model, SMLIR, uses MT to create a fully bilingual retrieval model. The documents are translated offline from language f to e and indexed as bilingual "pseudo-parallel" documents. At query time, the queries are translated from language e to f to create bilingual structured queries, which are used to retrieve the bilingual indexed documents. Experiments show that SMLIR returns more relevant results than monolingual models and previous bilingual models. We also compare several methods of translating queries, and find that using a Wikipedia-derived dictionary for NEs combined with an SMT dictionary works better than the SMT dictionary alone.

In the rest of the chapter, we give an overview of previous CLIR models, including closely related hybrid models. Then we introduce our SMLIR model. Finally, we present our experiments with Chinese-English CLIR and dicuss the results.

4.1 Prior Models

The main goal of CLIR is to allow users to access information in languages they do not know – that is, to retrieve *document language* results matching an information need expressed in the *query language*. As discussed in Chapter 2, CLIR evaluations are done in the document language in order to assess the retrieval accuracy in isolation from the effects of result translation.

The novelty of our approach lies in the assumption that users ultimately need to read the results in translation, i.e., that the document language results must be translated back into the query language. After all, if a user were proficient in the document language, he or she could simply enter the query in that language and perform a monolingual search. Even when users have some reading ability in the document language, browsing results in translation can be faster and more convenient than slowly reading them in a foreign language.

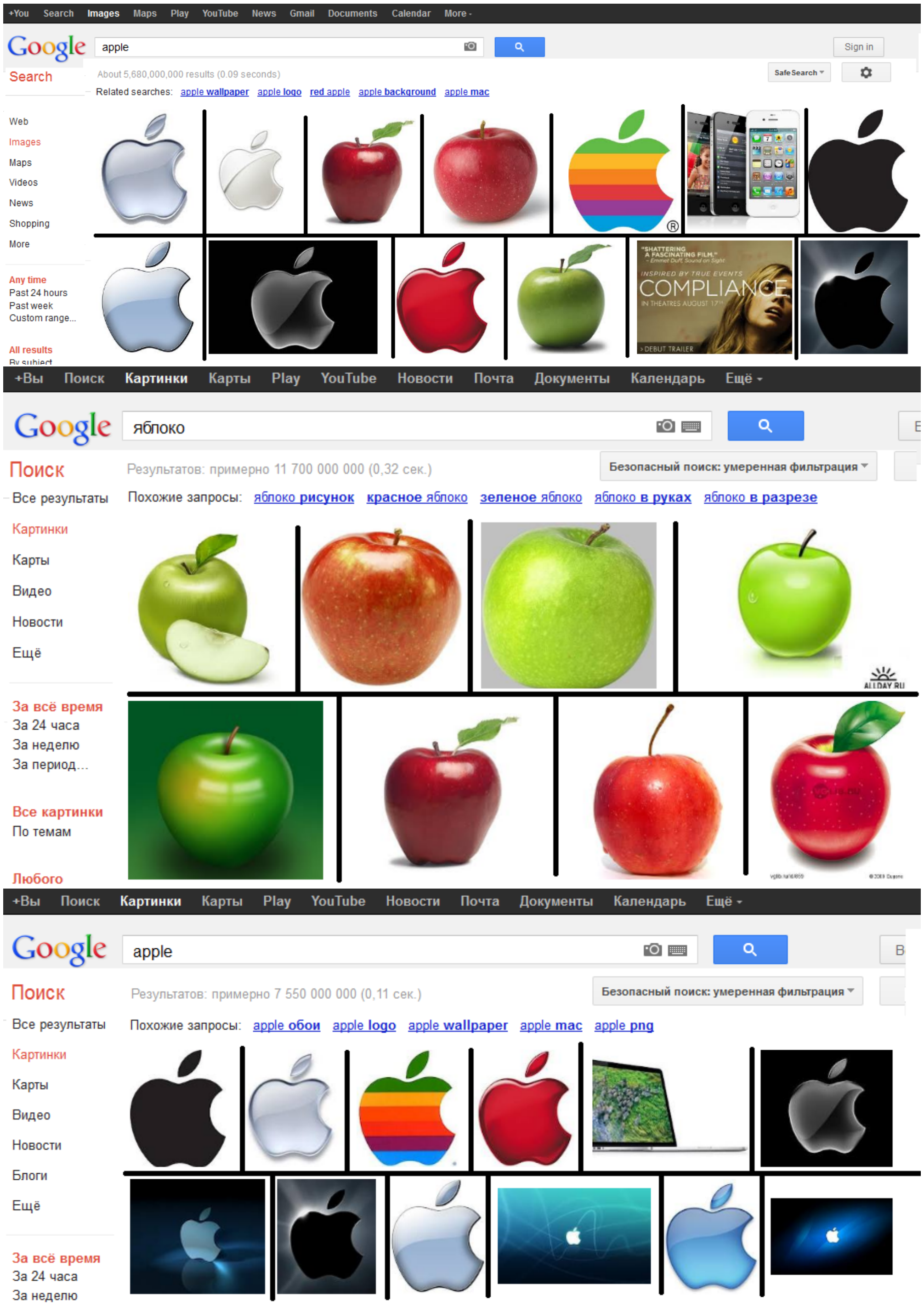
Once we assume that results will be translated back to the query language, we can leverage the translations to improve the retrieval model. How to leverage the translations depends on how the result translation is implemented: we can either use an online MT system to translate the search results at query time, or an offline MT system to translate the full corpus ahead of time. The CLIR model that we present in this chapter relies on having a full corpus translation, since at the time we did not have access to an online MT system fast enough to translate all results at query time. In Chapter 7, we present a related model that relaxes this assumption and only requires that the current search results be translated.

Both of our models are hybrid models that exploit query translation and document translation approaches to CLIR. In contrast, "most current research and development on CLIR use query translation [only] due to its high flexibility" [Nie, 2010]. In the following sections, we define and compare **document translation** and **query translation**, and describe additional **hybrid models** that combine the two.

4.1.1 Query Translation (QT)

Many CLIR systems use a **query translation (QT)** approach, in which the documents are indexed in their source language(s), the query is translated into each of the F document languages, and the retrieval is done completely in languages F. There has been extensive research comparing different approaches to query translation [Gao *et al.*, 2001; Wu *et al.*, 2008], which continues to the present day [Herbert *et al.*, 2011; Magdy and Jones, 2011]. Query translation can be done via manually-built dictionaries, MT, transliteration, or a combination of all three. QT faces many challenges in determining how to translate a short query with very little context, specifically:

- **Ambiguity before translation (in the query language)**: As in monolingual IR, polysemous query terms can lead to ambiguously expressed information needs. For instance, a user searching for "bass" may be looking for a musical instrument, a fish, or a shoe company. For ambiguous query terms in CLIR, more than one translation is possible. If the query is translated as the bass fish but the user was looking for the bass instrument, only irrelevant results will be returned, leading to a loss in recall. Figure 4.1 shows how the ambiguous English query "apple" returns different cross-lingual results depending on how it is translated into Russian.



apple

About 5,680,000,000 results (0.09 seconds)

Related searches: apple wallpaper apple logo red apple apple background apple mac

яблоко

Результатов: примерно 11 700 000 000 (0,32 сек.)

Безопасный поиск: умеренная фильтрация

Похожие запросы: яблоко рисунок красное яблоко зеленое яблоко яблоко в руках яблоко в разрезе

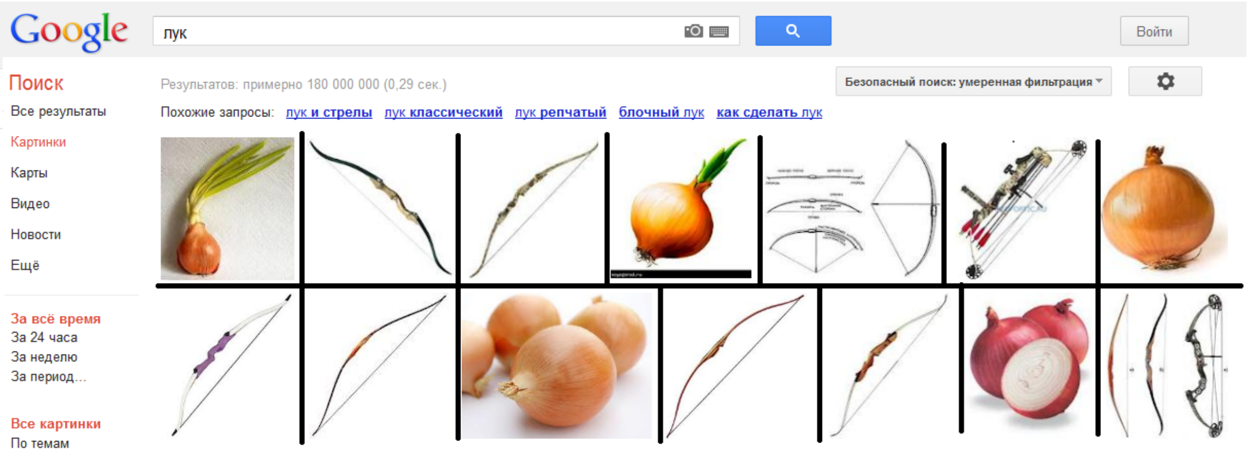
apple

Результатов: примерно 7 550 000 000 (0,11 сек.)

Безопасный поиск: умеренная фильтрация

Похожие запросы: apple обои apple logo apple wallpaper apple mac apple png

Figure 4.1: The search results for "apple" in English are ambiguous, as shown by the monolingual image search, which returns both apples and Apple logos. Cross-lingual image search results in Russian depend on how the query is translated: the query translation яблоко returns images of apples, while translating the query verbatim (as "apple") returns images of Apple logos or computers only.



лук

Результатов: примерно 180 000 000 (0,29 сек.)

Безопасный поиск: умеренная фильтрация

Похожие запросы: лук и стрелы лук классический лук репчатый блочный лук как сделать лук

Figure 4.2: An example of a query that becomes ambiguous after translation, in the document language. If the user does a cross-lingual image search for "onion" in English, and the query translation is "лук" in Russian, the user will see both relevant results for onions and irrelevant results for archery bows. The CLIR results are low precision even though the query was unambiguous in English.

- **Ambiguity after translation (in the document language)**: Even when a query term has an unambiguous one-to-one translation, a query term that is unambiguous in language e may be translated to an ambiguous or polysemous term in languages F. Figure 4.2 shows a query that is unambiguous in English (onion), but becomes ambiguous after translation into Russian (лук can mean onion or archery bow). If the user is searching for pictures of onions (the vegetable), the images of bows are not relevant, so the ambiguity leads to a drop in precision.

- **OOV terms**: For OOV words, there are no translations possible, so some back-off strategy must be used. In a cascaded query translation, multiple translation resources are consulted in succession, starting with the highest-precision resource. The lookup stops when a match is found, or when the top-k translations are found.

- **Untranslated terms**: In some cases, the correct query translation is no translation at all – that is, the query term should be translated as itself. Many company names, product names and some technical terms are written in English using Latin fonts, even in languages that use non-Latin alphabets. This can lead to a special case of query language ambiguity, where it is ambiguous whether to translate the term or not. In the example in Figure 4.1, an English query about apples (the fruit) should be translated into Russian, but a query about Apple computers should be left untranslated (in Latin script).

To deal with ambiguity, Pirkola [1998] introduced "structured queries" which use the term frequencies of all possible translations of a query term. Extensions address computational issues in computing frequencies (e.g., [Oard and Ertunc, 2002]) and augmenting the query translation with translation probabilities to weight each translated term [Darwish and Oard, 2003]. Language modeling has also been used as a basis for weighting term translations appropriately (e.g., [Xu and Weischedel, 2000; Lavrenko and Croft, 2001; Kraaij, 2004]).

Using weighted structured queries, the QT approach can represent distributions of all possible translations of the original query, and weight the retrieved results appropriately during ranking. Queries that are ambiguous in the query language (like "bass") can be represented as all possible meanings in the structured query. Similarly, synonyms and spelling variations can be easily incorporated into the structured query in order to increase recall. In additional to weighting each query term according to its translation probability, query translations from different sources (such as dictionaries or MT systems) can be weighted differently, depending on one's confidence in each resource. Using weighted structured queries, the QT approach can flexibly incorporate multiple meanings.

4.1.2 Document Translation (DT)

In the document translation (DT) approach to CLIR, all the documents in the full corpus are machine-translated and indexed in the query language e and monolingual searches are performed using the original query. Early work by Oard [1998] showed that DT outperformed QT, even when different methods were used to translate the query. In McCarley [1999], the relative performance of DT and QT depended on the quality of the MT system in each direction; in his experiment, the French-English MT system was better than the English-French MT system, so QT performed better for French queries and English documents, while DT performed better for English queries and French documents. Although these results suggested that DT has some advantages over QT, MT was very resource-intensive at the time, so there were few follow-up experiments.

Translating full sentences rather than short queries avoids some of the problems of QT because sentences tend to be grammatical and there is more context for translation disambiguation. But, as we saw in Chapter 3, MT systems still make translation errors. Figure 4.5 shows an example document translation that exhibits typical MT errors: Schwarzenegger's name could not be translated correctly, the sentence is ungrammatical, and an important word, "Chinese," has been deleted. This demonstrates the problem with a pure DT approach to CLIR that we saw in Section 3.2: although many queries contain NEs, names are especially difficult for MT to handle.

Foreign and rare names are especially problematic, especially since there are often several acceptable spellings of these names: our corpus contains at least three versions of Arnold Schwarzenegger's name in Arabic. Searching in less formal genres also requires handling name variations, nicknames and misspellings: various English documents in our corpus refer to Schwarzenegger as Schwartzenegger, Arnold, and the Governator. Unless the MT system used to translate the documents has perfect name translation and uniform spelling of name variations, a DT approach will always miss some documents. DT using MT also suffers from deleted tokens and so-called "hallucinated" insertions, both of which can hurt retrieval accuracy.

A further criticism of the DT approach is that it is infeasible due to resource constraints. Nie [2010] argues that "in a truly multilingual environment, one would desire to translate each document to all the other languages [and t]his is impracticable because of the multiplication of document versions and the increase in storage requirement." However, with current technology, storage is extremely cheap, so even in multilingual situations, DT is often appropriate. Institutions with multiple official languages such as the UN and the EU maintain fully parallel (human-translated) corpora that are indexed and searchable in all languages. If there is information that needs to be accessed in multiple languages that has not been translated by humans, it seems reasonable to use MT to translate it and index it, since MT is still less resource-intensive than human translation.

In some cases, fully multilingual DT is not possible: for instance, pre-translating the full Internet into all the languages on Earth is unrealistic. But it may still be useful to pre-translate the documents into one language, particularly if there is high-quality machine translation into this language. For example, many SMT systems are better at translating into English than other languages, simply because they have larger English language models and (often) more parallel data that includes English. Because of that, English is frequently used as a pivot or "bridge" language between two languages with very limited parallel data – for instance, to translate from

Maltese to Korean, the system first translates from Maltese into English, and then from English into Korean. In these cases, indexing documents in English may improve result relevance and use fewer translation resources at query time, since results only have to be translated from the pivot language into the query language.

4.1.3 Hybrid Methods

Both QT and DT attempt to map the query and the documents into a common language, but they use different translation strategies. In DT, each document is a large, coherent context with full, grammatical sentences, whereas a query may be short and non-grammatical, with little or no context, so it may be harder to translate. On the other hand, once a document is translated, any mistakes or deletions in the translation cannot be remedied, whereas translating a query allows for more flexibility in incorporating multiple possible translations using synonyms and related terms. Hybrid approaches to CLIR are based on the fact that DT and QT are complementary.

The merged approaches of McCarley [1999] and Chen and Gey [2003] for the joint use of QT and DT are most similar to our own. Chen and Gey do an "approximate" fast document translation by replacing each word in a document with the single most likely translation and subsequently build a query language index. McCarley uses a full MT system to translate his corpus, as we do in our system. In both systems, a document language search is done with the query translation over the indexed source documents (QT), and then a second query language search is done with the original query over the separately indexed translations (DT). Finally, the scores from the QT and DT runs are merged to get a score for the hybrid system, and the result documents are reranked. In both cases, the hybrid systems outperform the QT and DT systems; McCarley's hybrid system even outperforms his monolingual system, where human query translations are used to search the source documents.

In the rest of this chapter, we will refer to this hybrid approach of maintaining two separate indexes and merging the search results as the **merged approach**. We will compare our proposed approach to the merged approach in the experiments.

More recent research examines the joint use of query and document term translation [Wang and Oard, 2006]. They use bidirectional term alignments derived using Giza++ [Och and Ney, 2003] to translate both query terms and document terms. A key characteristic of their work is the mapping of translated terms to language specific synsets from WordNet [Miller *et al.*, 1990] for English and other languages. When a WordNet for a specific language does not exist, they compute the synsets for that language automatically from the Giza++ word alignments. This approach thus attempts to capture matches between query terms and document terms based on meaning. They experiment with a number of different methods for matching query-language synsets against document-language synsets. While these are similar to the issues we explore, we hypothesize that doing full document translations, rather than just translating document terms separately, allows for more accurate translation.

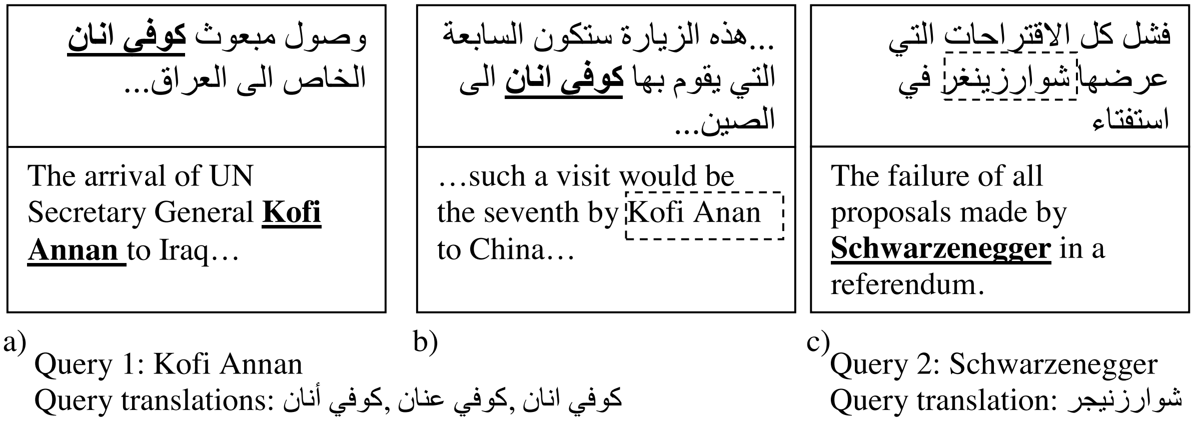
4.2 A New Model for CLIR: Simultaneous Multilingual IR

Our hybrid model, Simultaneous Multilingual IR (SMLIR), integrates document translation and query translation in a novel way that is particularly suited for CLIR with result translation. The approach exploits the fact that the corpus has been automatically translated to develop a hybrid model that integrates QT and DT into indexing and searching. A *pseudo-parallel document* is a document containing both the source (in $f \in F$) and MT (in e) of the document. The query is translated into each $f \in F$, and then a multilingual structured query is created, which represents the original query terms as well as their translations. Finally, the multilingual query is run over the pseudo-parallel indexed corpus to retrieve the results. In this model, the documents are searched simultaneously in both the document language, $f \in F$, and the query language e, thus allowing the relative advantages of the QT and DT approaches to complement each other.

The are two crucial differences between SMLIR and the previous hybrid models. First, SMLIR uses full document translation (rather than simple gisting or word-by-word translation), since it assumes that the results must be readable to end user. Not only does this mean that the document translations are better and more contextually correct, but it also means that the query translations and the document translations can use different, complementary approaches to translation. While the query translation may include results from MT, it goes beyond the top-1 translation to include a list of possible synonyms. In our model, we also include dictionary-based query translations.

The second main difference is that SMLIR is a coherent bilingual model, instead of a mixture of two monolingual models. Rather than build two indexes and run two separate searches, we build a single index where each document is indexed bilingually, as both the original document and its translation into the query language. Then we create a single multilingual query which combines the original query with the query translation(s) into the document language(s). Since we are not merging results from different IR systems (or indexes), no parameter tuning is required to determine whether to weight QT or DT higher or how to merge them.

Our approach is similar to the approach taken by [Nie and Jin, 2002] for multilingual information retrieval, where documents in multiple languages were combined in a multilingual index that could be searched with a single multilingual query which was constructed via query translation. In their system, the multilingual approach outperformed the "separate indexing then merging" approach. However, their goal was different than ours, since the documents were already multilingual and the purpose was to return relevant documents in multiple languages – the results were *not* translated back into the query language.



a)

وصول مبعوث **كوفى انان**
الخاص الى العراق...

The arrival of UN Secretary General **Kofi Annan** to Iraq…

b)

...هذه الزيارة ستكون السابعة
التي يقوم بها **كوفى انان** الى
الصين...

…such a visit would be the seventh by Kofi Anan to China…

c)

فشل كل الاقتراحات التي
عرضها شوارزينغر في
استفتاء

The failure of all proposals made by **Schwarzenegger** in a referendum.

Query 1: Kofi Annan
Query translations: كوفي انان, كوفي عنان, كوفي أنان

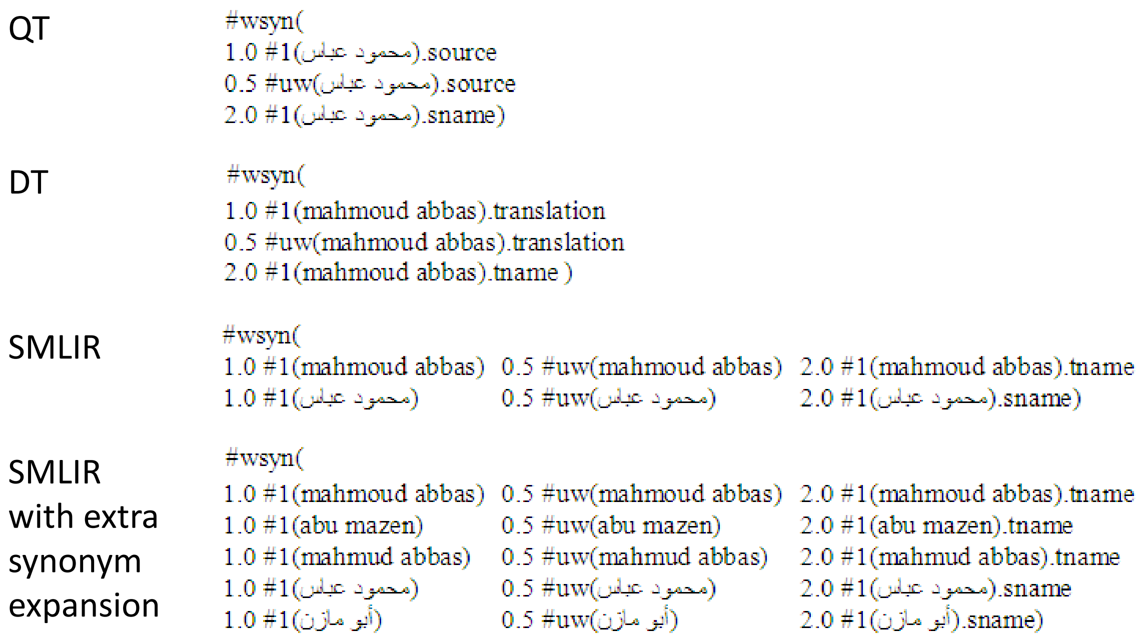
Query 2: Schwarzenegger
Query translation: شوارزنيجر

Figure 4.3: Indexed pseudo-parallel documents and multilingual queries. The documents are indexed using both the Arabic source (top half of the documents) and the English document translations (bottom half of the documents), and the queries are represented using both the English query and the Arabic query translations. Query matches are indicated with underlines; missed matches are indicated with dotted-line boxes. In document a, both QT and DT match. In document b, only QT matches due to a mistranslation (spelling error) in the document translation: Kofi Annan is translated as Kofi Anan. In document c, only DT matches because the query translation of Schwarzenegger does not match the spelling in the Arabic document. Using SMLIR, all three query-document pairs are matches.

4.2.1 SMLIR Examples

SMLIR attempts to mitigate the problems of DT and QT, and benefit from the advantages of each, by using both approaches simultaneously. Consider the pseudo-parallel documents and queries in Figure 4.3. For document a, both QT and DT succeed, and query 1 matches the document twice, once in the query language and once in the document language. However, document b translates Kofi Annan with a spelling error (Anan), so a system using only DT would not return this match for query 1. For document c and query 2, the QT system did not return this Arabic spelling of Schwarzenegger, but the MT system was still able to translate it correctly, so DT would return this match but QT would not.

By combining both methods in SMLIR, we are able to retrieve all three matches in Figure 4.3. Furthermore, documents like document a that match in both languages (via both QT and DT) are



```
QT          #wsyn(
            1.0 #1(محمود عباس).source
            0.5 #uw(محمود عباس).source
            2.0 #1(محمود عباس).sname)

DT          #wsyn(
            1.0 #1(mahmoud abbas).translation
            0.5 #uw(mahmoud abbas).translation
            2.0 #1(mahmoud abbas).tname )

SMLIR       #wsyn(
            1.0 #1(mahmoud abbas)  0.5 #uw(mahmoud abbas)  2.0 #1(mahmoud abbas).tname
            1.0 #1(محمود عباس)     0.5 #uw(محمود عباس)     2.0 #1(محمود عباس).sname)

SMLIR       #wsyn(
with extra  1.0 #1(mahmoud abbas)  0.5 #uw(mahmoud abbas)  2.0 #1(mahmoud abbas).tname
synonym     1.0 #1(abu mazen)      0.5 #uw(abu mazen)      2.0 #1(abu mazen).tname
expansion   1.0 #1(mahmud abbas)   0.5 #uw(mahmud abbas)   2.0 #1(mahmud abbas).tname
            1.0 #1(محمود عباس)     0.5 #uw(محمود عباس)     2.0 #1(محمود عباس).sname
            1.0 #1(أبو مازن)       0.5 #uw(أبو مازن)       2.0 #1(أبو مازن).sname)
```

Figure 4.4: Example structured queries for QT, DT and SMLIR in Indri.

Source	ادلي شوارزنجر بهذه الملاحظة هنا اليوم / الثلاثاء / في عشاء اقامته شبكة اعمال كاليفورنيا الامريكية ـ الصينية.
MT	He $wArznjr by these pointed out here today in a dinner banquet held by the network of California American.
Reference	Schwarzenegger made this statement here today, Tuesday, at a dinner held by the Chinese-American business association of California.

Figure 4.5: This document should be relevant to the query [Arnold Schwarzenegger], but it appears irrelevant due to mistranslation.

ranked higher than those that match in one language only (via only QT or DT). Matching in both languages means that we have additional, bilingual evidence that the query matches the document, and that the result will be translated correctly for the user. This is helpful for CLIR with result translation, where returning a bad translation can make a relevant document look irrelevant.

Figure 4.4 shows how the English query "Mahmoud Abbas" is represented using structured queries in the QT, DT and SMLIR approaches. (Indri query syntax is explained in further detail in Section 4.3.3.) The DT query is simply the original English query matched against the MT side of the documents (.translation). Ordered matches (#1) are weighted higher than unordered matches (#uw), and matches that are tagged as names in MT (.tname) are weighted the highest. The QT query is the 1-best query translation into Arabic matched against the source side of the documents (.source), with similar weightings on the source side of the pseudo-parallel document (the query specifies the .source and .sname fields instead of the .translation and .tname fields, respectively). The SMLIR query is a combination of the QT and DT queries, so it matches on both the source and translation fields of the indexed pseudo-parallel documents. The expanded SMLIR query includes three English variants and two Arabic variants for the same query. (If we had translation probabilities, we could weight each of the Arabic variants differently.)

4.2.2 Retrieval Model

A crucial feature of our hybrid system is that QT and DT are integrated into the retrieval model, rather than merged via parameter tuning as in the prior merged approaches ([McCarley, 1999; Chen and Gey, 2003]). Here, we show how our approach works in the Indri retrieval model (though the SMLIR approach is general and may be used with any IR system). Indri uses a query likelihood model for retrieval. For each document, Indri estimates $logP(Q|D)$, and then it ranks the documents according to the log probability. The model assumes term independence so that

$$logP(Q|D) = \sum_{q \in Q} logP(q|D)$$

where

$$P(q|D) = tf(q, D)/|D|$$

The latter equation represents the maximum likelihood estimate for individual term probabilities, which is simply the proportion of terms in D that are q. (To compensate for varying document length and to avoid the zero-probability problem, that estimate is adjusted with Dirichlet smoothing from the full corpus' statistics.)

For cross-lingual approaches, let D_f be the original document and D_e be its translation (similarly for Q_e and Q_f). For QT, q is replaced by its (optionally weighted) set of possible translations, $trans(q)$, meaning that:

$$P(q|D) = \sum_{w \in trans(q)} tf(w, D_f)/|D_f|$$

For DT, the estimate is done in the translation of the document:

$$P(q|D) = tf(q, D_e)/|D_e|$$

For SMILR, however, D_f and D_e are blended into a single representation D_{fe} and counts are estimated there, so

$$P(q|D) = \sum_{w \in trans(q)} tf(w, D_{fe})/|D_{fe}| + tf(q, D_{fe})/|D_{fe}|$$

Hybrid models used in earlier work (e.g., [McCarley, 1999]) average the QT and DT values. Ignoring the Dirichlet smoothing parameter, their merged approach is equivalent to:

$$P(q|D) = \sum_{w \in trans(q)} tf(w, D_f)/|D_f| + tf(q, D_e)/|D_e|$$

Note that if $|D_e| = |D_f|$ (or they differ only by a constant) and if no words in e occur in D_f and no words in f occur in D_e, then the two estimates differ only by a constant (since $|D_{fe}| = 2|D_e|$). That is, ignoring smoothing, the main theoretical difference between the two models arises when the source or translation document contains terms in the other language.

The ability to handle mixed-language text may be useful. We noticed that Chinese news articles tended to contain translations for Western names in parentheses after the first mention. Also, in our corpus, many blogs quoted English sources verbatim, but contained comments and discussion in Chinese or Arabic.[1] A pure query translation approach would miss these matches, since the query would not contain the original query terms.

In practice, the advantages of SMLIR go beyond mixed-language text because combining search results from different queries on different indexes is a difficult problem. In many IR systems, the scores per result are meaningful only for a particular query on a specific index. In other words, the scores are useful for relative ranking of search results for a single query, but cannot be used to compare results of different queries (or results from different indexes). The reason goes back to the constant that we ignored above. For queries of different lengths, the result scores have different normalization constants. Similarly, document length also affects the result scores.

Both query translation and document translation produce outputs of different lengths than the input. Furthermore, linguistic issues can significantly affect document and query length. Some languages have many more words per sentence than others. Tokenization also impacts length; for instance, Chinese can be indexed as single characters or using automatic segmentation. This means that running two separate queries over two separate indexes will yield two sets of results with scores that cannot be compared to each other. In our experiments, the scores of the DT and QT results sometimes differed by orders of magnitude from each other. Therefore, merging the results requires either ignoring the scores (and using ranks only), re-normalizing the scores or merging the scores via parameters, which must be tuned on a development set.

In contrast, the SMLIR model avoids this messy problem by having a unified bilingual model for both queries and documents. Only a single search is needed over a single index, and the result scores do not need to be re-normalized or re-ranked in an ad hoc manner.

In addition, translating from language e to language f may be easier than translating in the reverse direction. Therefore, one approach may perform better for a given query than another. Using the SMLIR model described above, the relative importance of QT and DT changes per query, depending on the relative quality of the query and document translations.

[1] It is difficult to measure exactly how often this happens without additional corpus annotation. Across several different Chinese corpora, we found that about 4% of Chinese source-language sentences contained Latin text, though not all of these cases matched the situations we described.

4.2.3 Practical Considerations

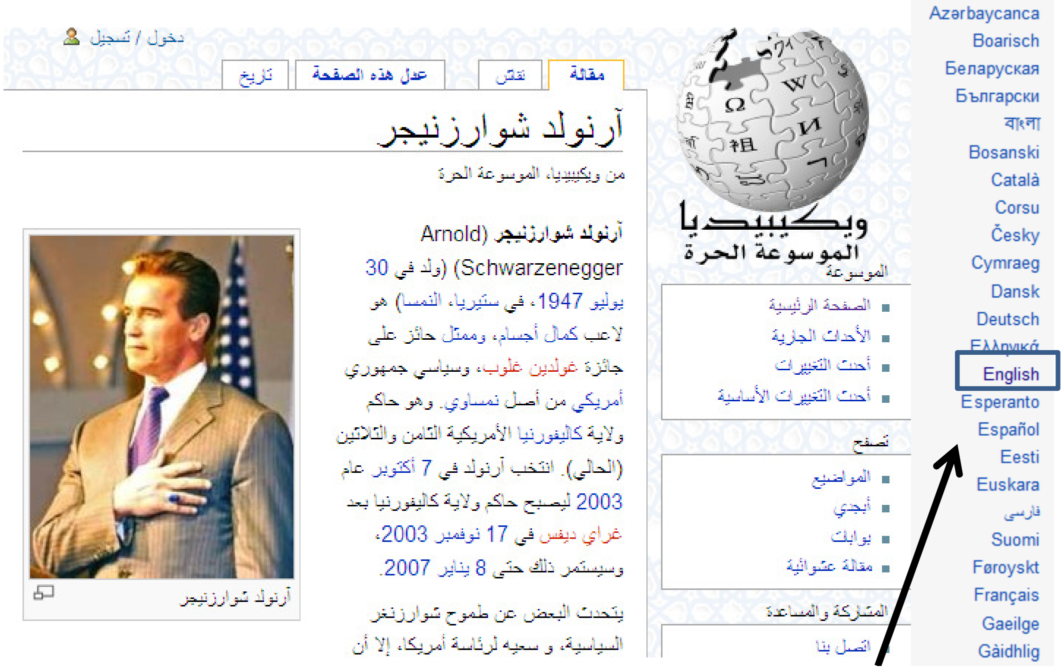
Although combining query translation with document translation has been shown to improve relevance [McCarley, 1999], CLIR systems typically do not use document translation since corpus translation is very resource-intensive. Alternatively, some systems use fast "approximate" document translation (e.g., [Chen and Gey, 2003; Oard, 1998]), where the result may be unreadable to humans but useful for IR. For the CLIR task, where the goal is to return documents in their source language, MT of a large corpus is a high cost with low reward. However, for our task of CLQA with result translation, where the goal is to return documents translated into the query language, corpus translation proves to be very useful. Since MT is typically resource-intensive, doing result translation at search time results in a trade-off between time and translation quality. Translating the documents ahead of time may allow for better translations, and also enables further offline analysis in the query language (for example, NE recognition).

From a practical point of view, joint indexing may also be simpler than a separate-indexing-then-merge approach. In the prior merged approach, two indexes are built and each query results in two separate queries, which then have to be merged. Using SMLIR, only a single index has to be managed, and each query results in one IR query, whose results can be returned without further processing. The query time for both methods is comparable.

4.2.4 Name-Centric Query Translation Using Wikipedia

We also explore various improvements to query translation, beyond the single-best machine translation used by McCarley [1999] and the approximate MT used by Chen and Gey [2003]. We derive synonym and translation dictionaries from Wikipedia. Since Wikipedia is created and edited by humans, we hypothesize that it will be better at translating certain terms than MT. Also, since Wikipedia is constantly being updated by users, it has very recent words that may not exist yet in the MT system's training data. In particular, Wikipedia contains the translations of many NEs, which may be mistranslated by MT systems (as described in Section 3.2). Ferrández *et al.* [2007] demonstrate that Wikipedia is an excellent source of NE translations for CLQA. For their Spanish-English CLQA application, a full 59% of the NEs should not be translated. Wikipedia helps them detect which ones to translate, as well as providing translations for many of the NEs. In contrast, in our case, we are dealing with languages that use non-overlapping alphabets (English, Chinese and Arabic) and thus the majority of NEs should be translated.

In order to build a high-precision NE-focused translation dictionary for English to Arabic and English to Mandarin, we took advantage of the user-created content in Wikipedia. In Wikipedia, each article may have links to the same (or similar) articles in other languages, as shown in Figure 4.6. We extracted these links to create a simple translation dictionary. (The links are not always bidirectional, so we extracted in both directions.) Users often add name translations in the first sentence of an article (e.g., "Mahmoud Abbas (Arabic: مَحمُود عَبَّاس)..."), so we extract those as well. Since this dictionary is derived from an encyclopedia, it contains many nouns and noun phrases, including many NEs. The name entries are biased towards famous NEs, and in particular, entities that are somehow notable in both languages. Unlike typical MT dictionaries, these translations are not exact, word-for-word translations; for instance, "Hillary Rodham Clinton" is a headword



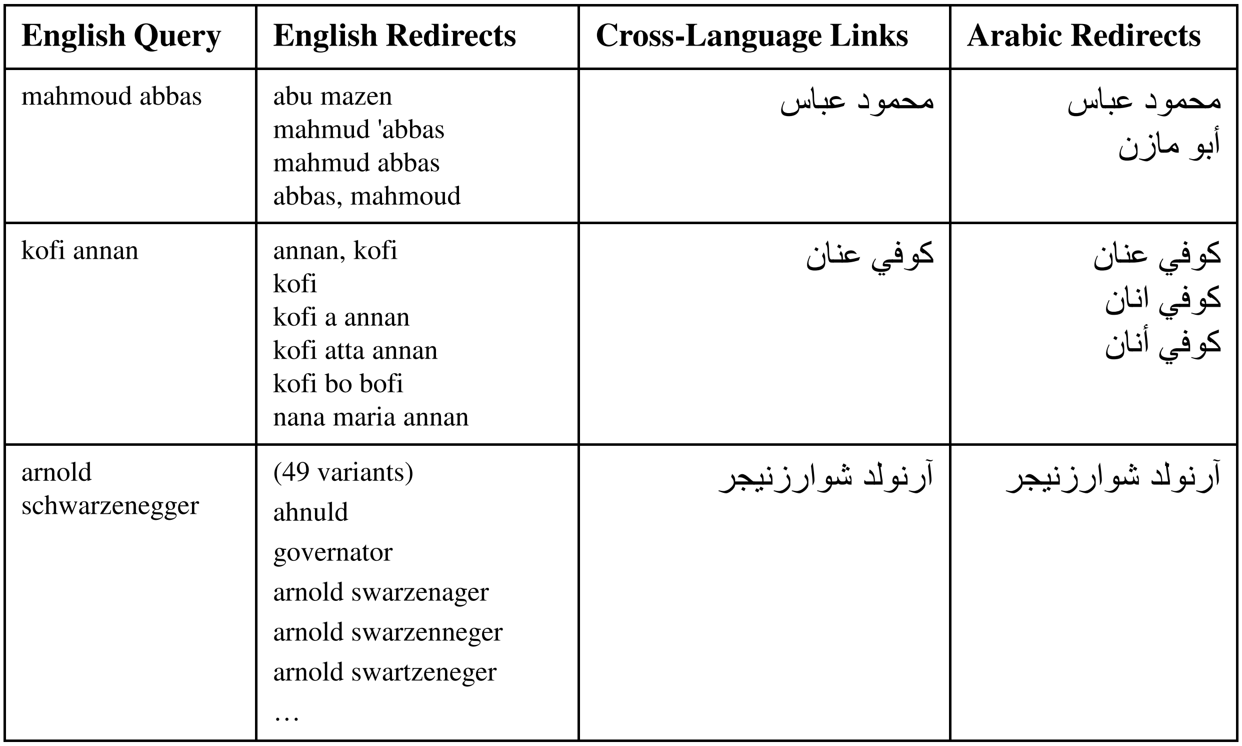
آرنولد شوارزنيجر

من ويكيبيديا، الموسوعة الحرة

آرنولد شوارزنيجر (Arnold Schwarzenegger) (ولد في 30 يوليو 1947، في ستيريا، النمسا) هو لاعب كمال أجسام، وممثل حائز على جائزة غولدين غلوب، وسياسي جمهوري أمريكي من أصل نمساوي. وهو حاكم ولاية كاليفورنيا الأمريكية الثامن والثلاثين (الحالي). انتخب آرنولد في 7 أكتوبر عام 2003 ليصبح حاكم ولاية كاليفورنيا بعد غراي ديفس في 17 نوفمبر 2003، وسيستمر ذلك حتى 8 يناير 2007.

يتحدث البعض عن طموح شوارزنغر السياسية، و سعيه لرئاسة أمريكا، إلا أن

Figure 4.6: The Arabic Wikipedia page for آرنولد شوارزنيجر (Arnold Schwarzenegger) links to the English page for Arnold Schwarzenegger. Even if it did not, the first sentence in Arabic lists the English name in parentheses. We extract bidirectional cross-language links as well as parenthesized translations in the first sentence of Wikipedia articles.



English Query	English Redirects	Cross-Language Links	Arabic Redirects
mahmoud abbas	abu mazen mahmud 'abbas mahmud abbas abbas, mahmoud	محمود عباس	محمود عباس أبو مازن
kofi annan	annan, kofi kofi kofi a annan kofi atta annan kofi bo bofi nana maria annan	كوفي عنان	كوفي عنان كوفي انان كوفي أنان
arnold schwarzenegger	(49 variants) ahnuld governator arnold swarzenager arnold swarzenneger arnold swartzeneger …	آرنولد شوارزنيجر	آرنولد شوارزنيجر

Figure 4.7: Synonyms and translations derived from Wikipedia for query translation. Note that there are two errors in the English redirects for Kofi Annan: "kofi bo bofi" and "nana maria annan."

in English, which links to the Arabic headword for "Hillary Clinton."

In addition to translation links, Wikipedia users can also add redirect links. These links match a name variation with the canonical form of an article title. For example, there is an English link that redirects the common misspelling "schwarzeneger" to "Arnold Schwarzenegger," and another for the slang term "governator." By aggregating all the redirects for a certain article, we can create sets of name variations from each version of Wikipedia; for our purposes, we extracted redirect sets from Arabic, Chinese, and English. It is important to note that these sets are not always synonyms, but may be related words, common misspellings, or even intentional spam. Since our corpus contained blogs and newsgroups, misspellings and slang were useful to us. Relying on user-generated content was important, since these variations would not normally be found in a standard dictionary; however, it may add noise to the dictionary.

Using the extracted translation dictionaries and redirect sets, we are able to look up a query term and get a set of variants in both languages. Figure 4.7 shows the results of looking up two queries. First, each English query is expanded by the English redirects list. Then, we use the translation dictionary to try to translate each redirect variant. Finally, we expand each translation in the Arabic redirects list. For "mahmoud abbas," the English list contains valid spelling variants and a common nickname ("abu mazen"), and the Arabic list contains "mahmoud abbas" and "abu mazen." However, for Kofi Annan, we have two errors in the English list: "Nana Maria Annan" is his wife, and "Kofi Bo Bofi" is the punchline to a joke. The Arabic list contains a translation of "Kofi Annan" and two spelling variants.

As is typical, there is a trade-off between precision and recall using the name variations (expansions). For example, the term "William Jefferson Clinton" is not present in our English-Arabic dictionary, but if we expand it to "Bill Clinton," we can find a translation. In Figure 4.7, both the Arabic Wikipedia and the English Wikipedia list "Abu Mazen," but there is no explicit link between them, so they are only available through the redirect lists. However, famous people may have tens or hundreds of redirect terms, so it may hurt precision to include them all.

4.3 Experimental Setup

Our evaluation was based on the DARPA GALE Y2 distillation evaluation, which was a template-based CLQA task with result translation. The task is described more in Section 2.2.2 and the templates are listed in appendix A. We focused on the CLIR component of the task only – that is, retrieving Chinese or Arabic documents in response to an English template-based query.

4.3.1 Queries

Overall, out of 17 templates, 10 have NEs as arguments, 3 have events or topics as arguments, and 4 contain both NEs and non-NEs as arguments. The arguments are typically short and name-centric: in GALE's second year, the average argument length was 3.4 words, and less than 10% of arguments contained no NEs.

The argument length is interesting, because the query arguments are more similar to short web queries than long TREC-like query narration/descriptions. Unlike web queries, GALE query arguments are well-formed grammatical units – they are always noun phrases. Given these characteristics, query translation is in some ways easier for GALE than for other tasks because we do not have to translate long sentences or analyze semantic roles. However, the context of a longer narrative is not available to guide query translation.

The GALE queries are non-factoid, template-based questions, for example, "WHERE HAS [Tony Blair] BEEN AND WHEN?" The filler text (in capitals) is used to frame the template, but does not supply useful query terms for querying the corpus. The argument is indicated by brackets, and is sometimes marked as a NE of a specific type. We ran a NE recognizer on the query to get more fine-grained markup. There are also optional slots for related words, name variants, and locations. Figure 4.8 shows a full XML query with two slots, one set of query term synonyms and one set of related words.

From the whole query, we extracted a set Q_e of English query phrases/words which contained all arguments, related words, locations and phrases marked as NEs. For example, the query argument "Osama bin Laden in Iraq" would generate three terms based on NE markup: "Osama bin Laden," "Iraq" and "Osama bin Laden in Iraq."

4.3.2 Corpus

The GALE Y2 distillation corpus was used for our CLIR experiments. The newswire (formal text) corpus contains approximately 100k news stories in each language. The webtext (informal text) corpus contains approximately 100k web postings in English and Chinese, with roughly 60k web postings in Arabic. The audio corpora contain approximately 40 hours of broadcast conversation (informal speech) in each language and 40 hours of broadcast news (formal speech) in each language.

The Arabic and Chinese documents were translated into English using the RWTH production MT system, described in Section 3.1.1. All documents were marked up with NEs (including co-reference resolution), time expressions, and ACE events in both the source and target languages. Character segmentation was used to index and search Chinese text.

4.3.3 Indri IR System

We used the v2.5 of the open-source Indri retrieval system for all retrieval experiments. Indri is a powerful retrieval system that combines inference networks and statistical language modeling approaches to retrieval [Metzler and Croft, 2004]. We chose Indri primarily for two reasons: (1) it provides a convenient mechanism for restricting queries to XML elements and (2) it provides a

```
<query template-number="6" query-id="10.2">
        <query-text>DESCRIBE THE RELATIONSHIP OF
                              Tung Chee-hwa TO Beijing</query-text>
        <query-arg arg-num="1" arg-type="Person">
                <arg-value>Tung Chee-hwa</arg-value>
        </query-arg>
        <query-arg arg-num="2" arg-type="Organization">
                <arg-value>Beijing</arg-value>
        </query-arg>
        <context>
        <equivalent-terms>
                <term>Beijing</term>
                <term>Chinese government</term>
        </equivalent-terms>
        <related-words>
                <term>political view</term>
                <term>political ideologies</term>
                <term>Hong Kong</term>
                <term>political influence</term>
        </related-words>
        </context>
</query>
```

Figure 4.8: A full GALE template-based query with two slots (Tung Chee-hwa and Beijing), one set of query term synonyms and one set of related words.

weighted synonym operator as part of the query language.

We use the XML operators to restrict query terms to matching in a single language. For example, the French query term “are” should not match an English document containing “are.” Although each document contains text in two languages, the source and translation are in separate XML elements, and we can query them separately using Indri. For example, it may be useful to restrict common nouns to matching in one language only, but match proper names in both languages.

Our queries used the following operators of the Indri query language:

- $\#1(abc)$ indicates that terms or features a, b, and c are a phrase and must appear in order and adjacent to each other.
- $\#combine(ab)$ indicates that the score associated with a document should be a combination of the scores of its operands. It is the default multi-term operator of Indri.
- $\#wsyn(w_a a w_b b w_c c)$ indicates that terms or features a, b and c should be treated by Indri as if they are the same term, and counts of the terms are weighted by w_a, w_b and w_c, respectively. We use this operator to incorporate probabilistic translations of words into the queries: alternate translations are listed with weights reflecting the estimated chance that the words are actually translations of the query term. That is, when calculating document or collection probabilities, the counts of all terms are added together and treated as one. Note that the synonym operator allows other operators such as $\#1()$ as a feature.
- $\#OP().field$ forces Indri to evaluate the operator only within the indicated field name. This feature allows us to run a query within the source text, the translated text, or both.

4.3.4 QT Dictionaries

We extracted cross-language translations and within-language synonyms (redirects) from Wikipedia as described in Section 4.2.4. As of January, 2008, the number of articles for each language was:

- English: 2,153,891
- Chinese: 159,392

- Arabic: 50,098

To compensate for the small Arabic Wikipedia, we combined it with a translation dictionary extracted from other Arabic dictionaries: a bilingual name dictionary extracted from the Buckwalter analyzer [Buckwalter, 2004] and a name dictionary extracted from ArabEyes.[2]

Although the queries in our CLQA task are name-focused, there are many non-names as well, such as topics and events. To translate non-names, we used a probabilistic word translation table derived from bidirectional word alignments extracted from GIZA++ [Och and Ney, 2003] by the MT members of our GALE team.

4.3.5 Annotation and Evaluation

In this chapter, we focus on improving the retrieval model independent of the translated results, so our evaluation is done in the document language, as is typical in CLIR evaluations. The judgments were done by 12 native speakers of Mandarin Chinese.[3] The documents could be judged *Relevant* or *Not Relevant* for each query. The queries were taken from the GALE Y2 distillation training set. We used 39 queries as development data and tested on 96 queries. We pooled the top 10 documents from each system for each query for annotation. In total, 13,942 query-document pairs were judged.

We report IR results using precision at 10 and NDCG at 10, both of which are standard IR metrics for ranked retrieval tasks [Manning *et al.*, 2008] and are defined in detail in section 2.2.3. Precision at k represents the average number of relevant results in the top k ranked results list. NDCG takes into account the relative ranking that each system gives to the returned documents, so two IR systems with the same precision could have different NDCG scores if one ranks the relevant results higher than the other. Significance on NDCG is done using a two-tailed t-test.

[2] Thanks to Nizar Habash for sharing this dictionary with us; it was extracted from the ArabEyes website (www.arabeyes.com).

[3] While the task included Arabic and Chinese, and our SMLIR system was implemented for both languages, we could not find enough Arabic annotators to evaluate the models in Arabic.

4.4 Results

We compared our approach, SMLIR, against document translation (DT) and query translation (DT) baselines as well as the previous hybrid model, merged.

- For the **document translation (DT)** baseline, the original query plus English expansions were matched against the indexed MT only.
- For the **query translation (QT)** baseline, a word-based probabilistic translation dictionary derived from MT was used in combination with a synonym and translation dictionary extracted from Wikipedia. The query was translated into Chinese and expanded, and matched against the indexed source text only.
- The **merged** system is a simple hybrid system (based on [McCarley, 1999]) that re-ranks results from the QT and DT baseline systems by their averaged, normalized Indri scores.
- The **SMLIR** system does a bilingual search, using translations and query expansions in both source and target languages, to match the full indexed pseudo-parallel bilingual documents.

In the first set of results, all systems used both Wikipedia and SMT dictionaries for query translation. In the second set of results, we compare the effect of different QT dictionaries on retrieval.

4.4.1 CLIR Evaluation

Figure 4.9 shows the NDCG at 10 results comparing all the CLIR models. For the monolingual models, document translation (DT) does significantly better than query translation (QT). The poor performance of QT is due to two problems: the prevalence of rare names in the queries, which were not covered by the translation dictionary, as well as some issues in translation of non-name phrases. A query translation module that included transliteration might improve performance of QT for names. For non-name arguments (such as "the cigarette smoking ban"), using full SMT rather than the typical approach of word-by-word translation might lead to better QT.

SMLIR outperforms all other systems significantly by NDCG at 10 (as shown in Figure 4.9), and Figure 4.10 shows the same trend for precision at 10. Surprisingly, the merged hybrid system does worse than DT, though at a lower level of significance (97.5%). It seems that for the merged approach, the poor performance of the QT system just serves to degrade the performance of the DT system. One of our initial criticisms of the merged system was that document scores are not comparable across queries, so combining them in any way is ad hoc. We found numerous examples of this in our error analysis. Sometimes scores from a QT query were orders of magnitude smaller or larger than scores from a DT query, in which case merged would end up favoring the system with the larger scores, rather than combining DT and QT in a more principled way, as we expected SMLIR to do.

Figure 4.11 compares NDCG and precision across different query types. All the retrieval models performed worse on queries with event/topic arguments. (The arguments are marked as events or topics, although the events may have NEs within them.) Whereas merged made significant gains over both QT and DT on event/topic queries, SMLIR barely outperformed them. On the other hand, for queries with name arguments, merged does worse than DT, and SMLIR does best.

4.4.2 QT Methods

We also performed experiments with various query translation methods. Since Wikipedia is an encyclopedia, we used it for translating NEs, which are problematic for machine translation. We compared translating the names only using Wikipedia against translating all terms using a probabilistic MT dictionary. For all settings that used Wikipedia, we expanded names in the query using the synonym lists derived from Wikipedia, and then translated all synonyms.

Surprisingly, just translating the names with Wikipedia did better than using the MT dictionary to do translation of all terms, as shown in Figure 4.12. This may be due to the fact that Wikipedia translations are typically high precision but low recall, whereas an MT dictionary typically contains many (weighted) translations, not all of which are appropriate for a given context. We expected that combining high-precision NE translation with a probabilistic translation dictionary would perform best, but combining the dictionaries did not improve the results further. In any case, the success of using Wikipedia highlights the importance of NE translation for our CLIR task.

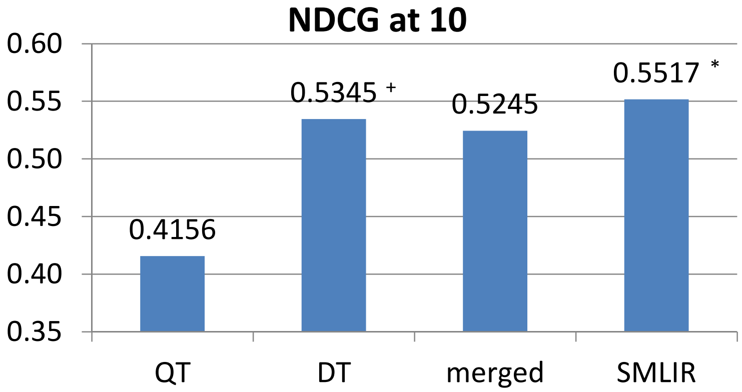


Figure 4.9: NDCG at 10 for the 96 test queries: * indicates 99% confidence, + indicates 97.5% confidence using a two-tailed t-test.

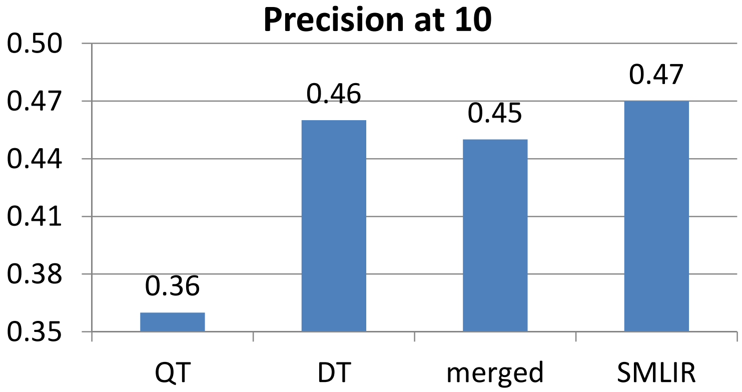


Figure 4.10: Precision at 10 for the 96 test queries.

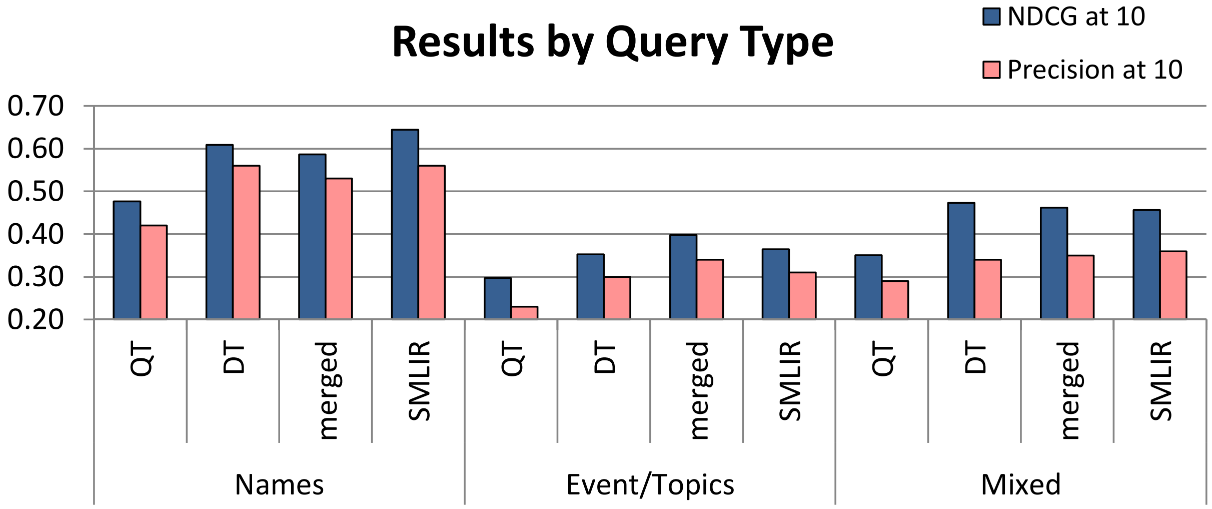


Figure 4.11: The CLIR evaluation results, split into queries with NE parameters, queries with event/topic parameters, and mixed queries with both types of parameters.

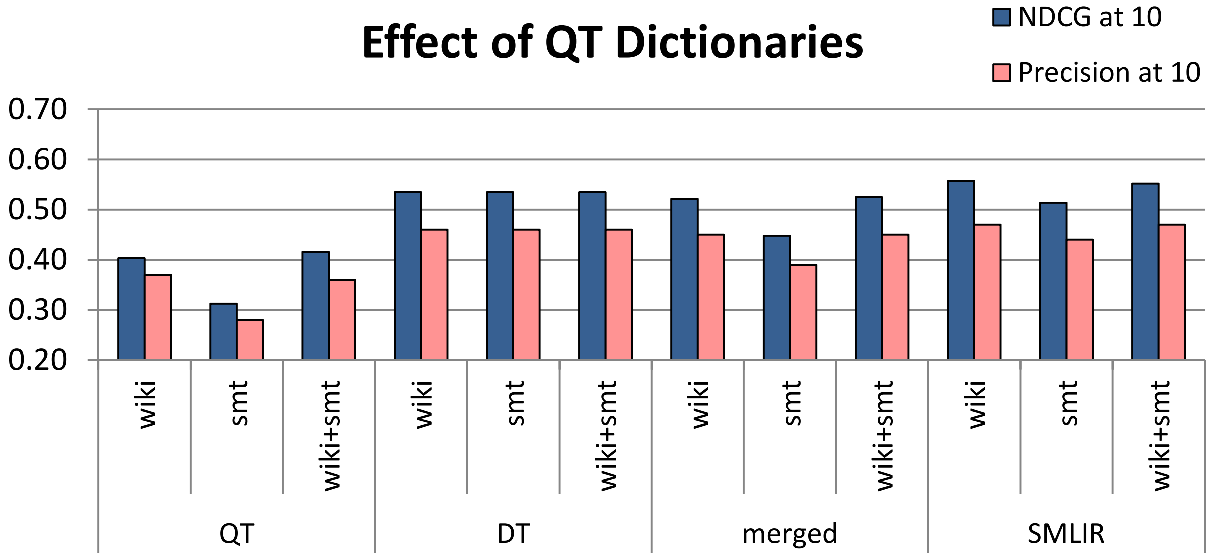


Figure 4.12: The effect of using different QT dictionaries on NDCG and precision at 10.

4.5 Conclusion

In this chapter, we described a CLIR model designed to be used as a subtask of CLQA with result translation. When considering a real-world application of CLIR, where results would have to be translated back into the query language, it may be necessary to do full translation of the corpus ahead of time. The simultaneous multilingual IR (SMLIR) model was designed to exploit this extra target-language information.

SMLIR uses bilingual information in both the indexed pseudo-parallel documents and the multilingual structured queries that are issued over the index. This novel hybrid model mitigates the problems with QT and DT by using the complementary advantages of each approach: QT allows for flexibility by integrating multiple translations of the query into the structured query, while DT may produce better translations due to a larger context. Since we used different approaches for translating documents (full SMT) and translating queries (a combination of SMT and manual dictionaries), searching bilingually allows for the possibility of matching in either language, while promoting matches that occur in both languages.

Previous hybrid approaches had maintained separate indexes for DT and QT and then merged the results. However, merging search results from two different queries over two different indexes is challenging because the result scores from different searches are typically not comparable. In contrast to these "merged" approaches, SMLIR utilizes a unified bilingual representation for both queries and documents. Since SMLIR does a single search over a single index, it does not have to use an ad hoc result merging strategy.

The SMLIR model was motivated by the failures of a pure DT approach that we documented in our error analysis in Chapter 3. In that experiment, MT errors during document translation prevented many relevant documents from being retrieved at query time because the search was done in the document language only. SMLIR addresses this problem by simultaneously searching in both the document and query languages, so even documents that are garbled by MT can be retrieved.

The error analysis in Chapter 3 also highlighted the importance of named entities in our CLQA task. While NE translation is crucial for retrieving relevant results and understanding result translations, NEs are particularly difficult for MT systems to translate. In this chapter, we showed how specialized high-precision NE dictionaries can improve query translation over using an SMT phrase table alone. In future chapters, we extend this idea and use the NE dictionaries in conjunction with NE recognition to adapt task-agnostic MT system output to the needs of our CLQA task.

The improved recall of SMLIR (over the DT model) actually leads to another problem when search results are presented to the user. The sentences that SMLIR finds that DT was unable to find are those that have MT errors. However, once retrieved, these relevant results may appear irrelevant to the user because of the MT errors. In Chapter 5, we show how we can use the SMLIR model to detect certain types of MT errors in the document translations, and in Chapter 6 we discuss how to automatically post-edit the translations to correct the errors.

Chapter 5

Automatically Detecting MT Errors

The error analysis in Chapter 3 showed how using a document translation approach to CLQA was very vulnerable to errors in MT: mistranslations during corpus MT frequently led to recall errors, where relevant documents could not be retrieved by the CLIR system. The SMLIR model addressed these recall errors by integrating query translation and document translation, so that even documents with MT errors could be retrieved. This is useful in the CLIR task, where the goal is to retrieve source-language documents. But in CLQA with result translation, a relevant result with MT errors may appear irrelevant, and therefore be no more helpful to the end user than an irrelevant result. Consider the example in Figure 5.1: the Arabic sentence is relevant to the query, but after MT, the English sentence appears irrelevant because the phrase "Pyong Yang" (the capital of North Korea) is lost in translation.

In this chapter and the next chapter, we explore ways to automatically detect and correct MT errors that impact the CLQA task so that sentences that are relevant in the document (source) language are *understandably relevant* after MT. We consider the problem of automatic post-editing of MT independent of the larger CLQA task, just as in the previous section we analyzed CLIR in isolation. (In Chapter 7, we evaluate our techniques in the context of the full task of CLQA with result translation.)

Specifically, given an MT sentence, we wish to answer the following questions:

1. Does the sentence have MT errors that would prevent the user from understanding it? In our error analyses in Chapter 3, the MT errors that had the largest impact on translation

adequacy were NE mistranslations, missing content words and mistranslations. In this chapter, we present algorithms for identifying these errors on the phrase level without reference translations.

2. For each phrase-level error, how should the phrase have been translated? In the next chapter, we describe our methods for finding phrase translations – even when the MT system could not – using techniques from CLIR and resources from the wider CLQA task.

3. Given a phrase-level error and a list of possible translations, how can we edit or rewrite the MT sentence to correct the error? This step builds upon the output of the previous two steps, and we explore it in detail in the next chapter.

First, we present related work on reference-free evaluation and quality estimation in MT. Then

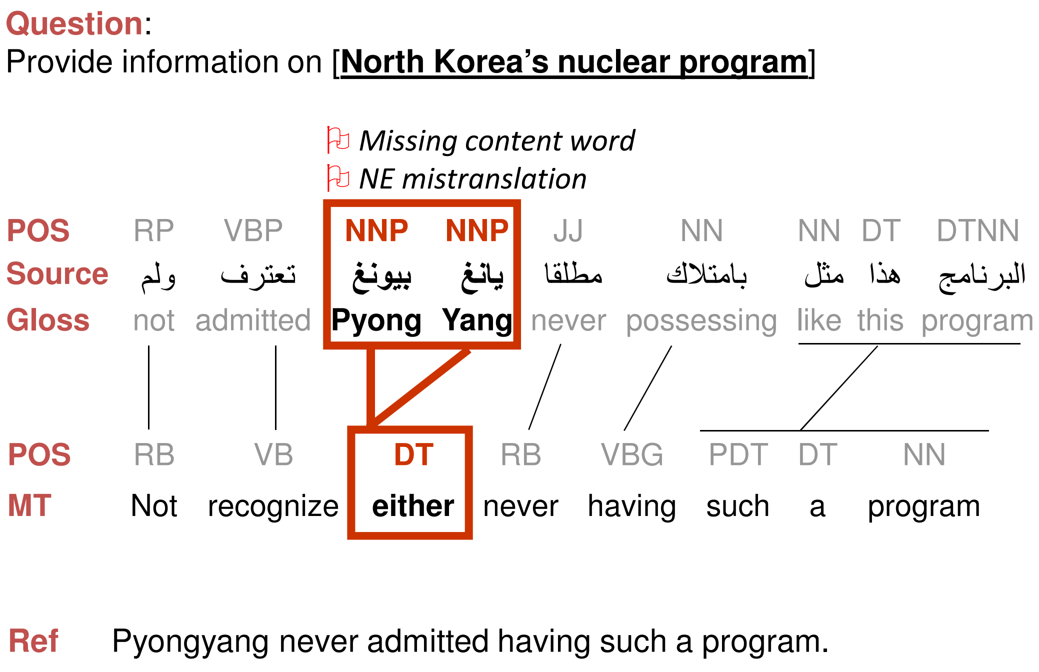

Question:
Provide information on [North Korea's nuclear program]
Missing content word
NE mistranslation
POS RP VBP NNP NNP JJ NN NN DT DTNN
Source ولم تعترف بيونغ يانغ مطلقا بامتلاك مثل هذا البرنامج
Gloss not admitted Pyong Yang never possessing like this program
POS RB VB DT RB VBG PDT DT NN
MT Not recognize either never having such a program
Ref Pyongyang never admitted having such a program.

Figure 5.1: This sentence is relevant to the query in Arabic, but not in MT. The pseudo-parallel sentence consists of the Arabic (source), Arabic POS, English (MT), English POS and the MT alignments between the source and MT. The selected phrase was flagged as an NE mistranslation and a missing content word error.

we describe our main contribution: lightweight, language-independent algorithms for identifying phrase-level MT errors. Finally, we experiment with using error detection to improve MT output without automatic post-editing. Instead, we integrate our error algorithms into a two-stage SMT system and evaluate it within the context of CLQA.

5.1 Related Work

There is extensive prior work in describing MT errors, but they usually involve post-hoc error analysis of specific MT systems (e.g., [Kirchhoff *et al.*, 2007], [Vilar *et al.*, 2006]) rather than online error detection. Errors involving missing content words are especially detrimental to translation adequacy, and prior work has shown that these error occur across different language pairs and different types of SMT systems. Zhang *et al.* [2009] described how errors made during the word-alignment and phrase-extraction phases in training phrase-based SMT often led to spurious insertions and deletions during translation decoding. Condon *et al.* [2010] found that 26% of their Arabic-English MT errors were verb, noun or pronoun deletions. Similarly, Vilar *et al.* [2006] found that 22% of Chinese-English MT errors were content deletion. Popović and Ney [2007] reported that 68% deleted tokens from their Spanish-English MT system were content words. In our own work on a Chinese-English cross-lingual semantic role labelling task (the 5W task), we found that content word deletion during MT accounted for 17-22% of the errors on this task [Parton *et al.*, 2009].

Hermjakob *et al.* [2008] studied NE translation errors and integrated an improved on-the-fly NE transliterator into an SMT system. Other research focuses on detecting errors in MT based on comparison with reference translations [Zeman *et al.*, 2011]. This is useful for diagnosing MT system errors, but cannot be used in an MT application where reference translations are unavailable.

Other papers focus on detecting MT errors in the context of different cross-lingual tasks. Ji and Grishman [2007] detected NE translation errors in the context of cross-lingual entity extraction, and used the task context to improve Chinese-English entity translation. Ma and McKeown [2009] investigated verb deletion in Chinese-English MT in the context of CLQA. They tested two SMT systems, and found deleted verbs in 4-7% of the translations. After using post-editing to correct the verb deletion, QA relevance increased for 7% of the sentences, showing that an error that may have little impact on translation metrics such as BLEU [Papineni *et al.*, 2002b] can have a significant impact on cross-lingual applications.

Quality estimation is an area that is closely related to error detection and has been the focus of much recent work, spurred by a new shared task at WMT '12 [Callison-Burch *et al.*, 2012]. The goal of quality estimation is to predict sentence-level translation quality without reference translations (unlike MT evaluation, which often requires reference translations).

Many approaches try to learn which features make a good translation and then predict the quality of new translations without comparing to reference translations. Initially, research focused on using machine learning to distinguish machine translations from human translations [Corston-Oliver *et al.*, 2001; Kulesza and Shieber, 2004; Gamon *et al.*, 2005]. Later work tried to explicitly compute a confidence measure for each translated sentence using bilingual features [Quirk, 2004; Albrecht and Hwa, 2007]. More recent work, including the shared task at WMT12, has focused on ranking MT sentences.

TrustRank [Soricut and Echihabi, 2010] ranks the quality of translations from good to bad. More relevant to our work is [Specia *et al.*, 2011], who use confidence estimation to predict translation adequacy. They emphasize the importance of NEs for adequacy, and develop special features to estimate the probability of NE translation accuracy. Mehdad *et al.* [2012] also focus on estimating MT adequacy by doing cross-lingual textual entailment.

The main distinction between automatic error detection and quality estimation (or confidence estimation) is that error detection seeks to identify specific, phrase-level MT errors, whereas quality estimation attempts to score or rank translated sentences. Some approaches to confidence estimation do provide scores on the word- or phrase-level [Ueffing and Ney, 2005; Raybaud *et al.*, 2011]. These differ in several respects from our work. They assume that all tokens in a sentence can be judged correct or incorrect. Depending on how the judgments are done, this can be unrealistic. Comparing the surface forms to a reference translation as in [Ueffing and Ney, 2005] is overly harsh and penalizes synonyms (as BLEU does). Collecting word-level correct/incorrect judgments from humans (as in [Raybaud *et al.*, 2011]) is better, but still does not distinguish between completely incorrect translations (such as the word "laughed" being translated as a comma) and near-misses (translating "many" as "some"). Also, whether a translated word is correct or incorrect is highly dependent on how other words in the sentence are translated.

Another difference is that error detection seeks to flag translated phrases with specific types of errors, whereas confidence estimation typically groups all kinds of errors together as "incorrect" translations. In task-based MT, certain types of errors may be more harmful than others: for CLQA, we focus on adequacy, but for another application, fluency might be more important. For these applications, labelling the type of error is particularly useful.

5.2 Algorithms

The error detection algorithms take as input the pseudo-parallel sentences that we introduced in the previous chapter. Figure 5.1 shows an example of a pseudo-parallel sentence, which consists of:

- The original sentence in the *document language*, which for the purposes of MT is the *source language*. The source sentence S consists of one or more source-language tokens s_i.
- The MT side of the pseudo-parallel documents in the *target language*, which is also the *query language*. The target sentence T consists of one or more source-language tokens t_j.
- Word and phrase alignments from the MT system that specify which target-language tokens were generated from each source-language token. An alignment a_{ij} connects source token s_i to target token t_j. Each $t_j \in T$ is aligned to one or more $s_i \in S$, while each $s_i \in S$ may have a null alignment $\emptyset$ or be aligned to one or more $t_j \in T$.
- Additional NLP annotations are available in both languages. For detecting NE mistranslations, we assume that we have NE recognizer markup, and for detecting content word deletions, we assume part of speech (POS) tags. Other types of error detectors could benefit from additional kinds of markup, such as parses, event detection, NE co-reference, or sentiment analysis.

For each pseudo-parallel input sentence, an error detection module should produce as output a list of zero or more detected errors. Each error err is a pseudo-parallel phrase from the sentence consisting of:

- One or more source-language tokens S_{err} and their associated annotations.
- Zero or more target-language tokens T_{err} and their associated annotations.

- The word alignments between the source- and target-language tokens (if any), i.e., all alignments a_{ij} such that $s_i \in S_{err}$ and $t_j \in T_{err}$.
- A label describing the type of error detected, and any meta-data added by the error detection algorithm.

We describe several error detection modules, each of which targets a different kind of error that impacts translation adequacy. Note that since the error categories are not disjoint, the same error may be detected as multiple kinds of errors: e.g., an NE that is translated to a function word counts as both an NE mistranslation and a missing content word. These algorithms are lightweight, language-independent and MT-system-independent. The only requirement is that the MT system produces word or phrase alignments.

5.2.1 Named Entity Mistranslations

The error analysis in section 3.2 showed that, while named entity (NE)s are crucial to our CLQA task, they are especially difficult for MT systems to translate correctly. In the context of our task, we have additional knowledge that is unavailable to the MT decoder during document translation:

- **Bilingual NE annotations**: We run named entity recognition (NER) systems on both the source and target sentences. Most MT systems do not accept NE markup in their input and do not have special handling for NEs.
- **High-precision NE dictionaries**: In the previous chapter, we described how NE dictionaries mined from Wikipedia were significantly better at translating queries with NEs than MT dictionaries alone. It is non-trivial to incorporate manually built dictionaries into MT phrase tables (Vogel *et al.* [2003], Och and Ney [2000] and Habash [2009] describe several methods for doing so) because the dictionaries typically do not have translation probabilities. Dictionaries may also be domain-specific. In this case, the NE dictionary is helpful when we are certain that we are translating a NE (the "name domain"), but if it were added to the general MT phrase table, it could hurt the translations of non-NEs. For instance, the Arabic word حرب should be translated as "war" unless it is a person name, in which case it should be "Harb." Thus, the NE dictionary is most useful in conjunction with NE annotations.

- **The CLQA query**: If we are detecting errors in the context of CLQA, we have the query that was used to retrieve the current pseudo-parallel sentence, which often contains one or more NEs, sometimes with spelling variants.

We can exploit these knowledge sources to identify source-language NEs that have been translated incorrectly by the MT system in two ways: a query-independent method that applies to any NE in any sentence, and a query-specific approach that can be used for NEs present in CLQA queries. While the latter method applies to much fewer sentences, the sentences that are affected are likely to be more important to the task since they contain query terms.

(Query-independent) NE mistranslation detection: A source-language NE that is aligned to a target-language non-NE is an indication of a possible error. It could either be a mistranslation or an error in the NE recognition: either a false positive in the source language or a false negative in the target language. If the high-precision NE dictionary contains the source-language NE and it lists the aligned phrase as a translation, it is an NE recognizer error and should not be flagged as an error. Otherwise, it should be flagged as an NE mistranslation error. Algorithm 1 shows the steps carried out by this error detector.

This method is dependent on the accuracy of both the source- and target-language NE recognizer as well as the coverage of the NE dictionary, all of which will vary for different language pairs and different applications. If the source-language NE recognizer has higher recall than the target-language recognizer, many NEs will appear to be translated into non-NEs, but they will not be mis-categorized as errors if they are listed as entries in the dictionary. In our experiments on Arabic-English MT, the English NE recognizer is typically much more accurate than the Arabic one (though it is highly dependent on correct capitalization), so we frequently observe phrases that are not marked as NEs being translated into NEs. These phrases can also be checked against the NE dictionary (shown in the optional second loop in Algorithm 1): if the source phrase is in the dictionary but the MT phrase is not, it should be flagged as a possible error.

Query-specific NE mistranslation detection: If the error detection is done in the context of CLQA, the SMLIR model can help identify potentially incorrect translations in a sentence when a document-language match is found for a query but a query-language match is not found. The steps for query-specific error detection are listed in Algorithm 2. Consider the query "Provide information on [Pyong Yang]." Assuming the query translation includes the correct translation

Cause	Source	MT	Gloss	Detected?
Translation model deletion	تاليس	(empty)	Thales	Yes
Word mistranslation	تحدث	there	spoke	Yes
Phrase partial translation	الاستفادة من	to	to benefit from	Yes
Phrase mistranslation	ترجمة آلية	machine	machine translation	No

Table 5.1: Four ways that content words can get lost in translation. The detection algorithm covers the first three cases, but not the fourth.

بيونغ يانغ, SMLIR could find the sentence in Figure 5.1. This sentence is relevant in Arabic, but a user is unlikely to rate it relevant in English given the poor translation.

The errors detected by the query-specific error detection are particularly important because they occur in phrases containing the query terms. These phrases are crucial for understanding result relevance, so mistranslating them may make a relevant sentence appear irrelevant to the query. Detecting and correcting these errors would have an impact on CLQA precision, while fixing errors in results that are irrelevant to begin with would have no effect on the end task.

Query-specific error detection relies on the fact that the query translation may succeed even when the document translation fails. This is because the query translation module goes beyond single-best MT output to include multiple translations from a wide variety of resources. Query translation may have another advantage if MT is better in one direction than another (e.g., Arabic-English MT may be better than English-Arabic MT).

5.2.2 Missing Content Words

The error analyses in Chapter 2 showed that missing content words can significantly degrade translation adequacy, which is essential to the CLQA task. Content words can get dropped during translation in several ways, as shown in Table 5.1.

The simplest case of content word deletion is a complete deletion by the translation model – in other words, a token was not translated. We assume the MT system produces word or phrase alignments, so this case can be detected by checking for a null alignment. However, it is necessary to distinguish correct deletion from spurious deletion. Some content words do not need to be

Algorithm 1 The query-independent algorithm for detecting phrase-level NE mistranslations in a pseudo-parallel sentence.

S and T represent the source- and target-language sentences, respectively.

```
function DetectNEErrors(S,T)
    for all source-language NE phrases $S_{NE}$ do
        $T_{NE} \leftarrow$ target-language phrase aligned to $S_{NE}$
        if $inNEDict(S_{NE})$ AND $!isNETranslation(S_{NE}, T_{NE})$ then
            $addError(S_{NE}, T_{NE})$
        end if
    end for
    ▷ The second loop is suitable only if target-language NER is better than source-language NER
    for all target-language NE phrases $T_{NE}$ do
        $S_{NE} \leftarrow$ source-language phrase aligned to $T_{NE}$
        if $inNEDict(S_{NE})$ AND $!isNETranslation(S_{NE}, T_{NE})$ then
            $addError(S_{NE}, T_{NE})$
        end if
    end for
end function
```

Algorithm 2 The query-specific algorithm for identifying NEs from the query that are mistranslated in the document translations.

S and T represent the source- and target-language sentences, respectively.

Q represents the target-language query.

QT represents all the source-language query translations associated with query Q.

```
function DetectNEErrorsUsingQuery(S,T,Q,QT)
    for all source-language NE phrases $S_{NE}$ do
        if $S_{NE} \in QT$ then
            $Q_{NE} \leftarrow$ target-language NE from query $Q$
            $T_{NE} \leftarrow$ target-language phrase aligned to $S_{NE}$
            ▷ If the MT matches the NE in the query $Q$, it is not an error
            if $fuzzyMatch(Q_{NE}, T_{NE})$ then
                continue
            end if
            ▷ If the translation is in the NE dictionary, it is not an error
            if $isNETranslation(S_{NE}, T_{NE})$ then
                continue
            end if
            $addError(S_{NE}, T_{NE})$
        end if
    end for
end function
```

translated – for example some Arabic auxiliary verbs (such as كان "to be") are often correctly deleted when translating into English.

A more subtle form of content word deletion occurs when a content word is translated as a non-content word. This can be detected by comparing the parts of speech of aligned words. In Table 5.1, this occurs when the verb (تحدث "spoke") is translated as the expletive "there."

Another case of content word deletion occurs when a content word is translated as part of a larger MT phrase, but the content word is not translated. In the third example in Table 5.1, an Arabic phrase consisting of a noun and preposition is translated as just the preposition "to" instead of the phrase "to benefit from."

The fourth type of missing content word is the most difficult to identify. If a phrase containing a content word is translated into another phrase with a content word, it may or may not be a complete translation. For instance, if "spoke loudly" is translated as "yelled," it is a valid translation because the verb "yelled" includes the semantics of the adverb "loudly." On the other hand, if "spoke loudly" is translated as "explained," the adverb "loudly" is a missing content word. Detecting this kind of error is as hard as detecting a word-to-word lexical mistranslation, and we do not address these types of errors in this thesis.

The latter three kinds of content word deletion are considered mistranslations rather than deletions by the translation model, since the deleted source-language token does produce one or more target-language tokens. However, from the perspective of a cross-lingual application, there was a deletion, since some content that was present in the original is not present in the translation. For clarity, we refer to all of the cases in Table 5.1 as *missing* content words rather than deleted content words.

The deletion detection algorithm is motivated by the assumption that a source-language phrase containing one or more meaning-bearing words should produce a phrase with one or more meaning-bearing words in the translation. (Phrase refers to an n-gram rather than a syntactic phrase.) Note that this does not assume a one-to-one correspondence between content words – for example, translating the phrase "spoke loudly" as the single word "yelled" satisfies the assumption. The exact definition of "meaning-bearing word" will depend upon the language and POS tagset.

The error detection algorithm (shown in Algorithm 3) takes as input a pseudo-parallel sentence. The system iterates over all content phrases in the source sentence, and, for each word, checks whether it is aligned to one or more content words in the target sentence. If it has no alignment (line 7), or is aligned to only function words (line 12), the system reports an error. This rule-based approach has poor precision because of content words that are correctly deleted. For example, in the sentence "I am going to watch TV," "am" and "going" are tagged as verbs, but may be translated as function words, such as a future tense marker. To address this, we heuristically filtered frequent content words by using source-language IDF (inverse-document frequency) over the QA corpus. The cut-off was manually tuned on a development set.

This algorithm generalizes [Ma and McKeown, 2009] in several ways. First, it detects any deleted content words, rather than just verbs. The previous work only addresses complete deletions, where the deleted token has a null alignment, whereas this approach finds cases where content words are mistranslated as non-content words. We also relax the assumption that translation preserves part of speech (i.e., that verbs must translate into verbs), assuming only that a phrase containing a content word should be translated into a phrase containing a content word, which handles cases like verb nominalization (where a verb gets translated into a noun).

5.2.3 Other Errors

Finally, we describe two miscellaneous MT errors that are straightforward to detect.

Numbers: Numerals in sentences represent important pieces of information, but are not always handled well by MT systems. For instance, the error analysis in Chapter 3 showed that 6% of numerals in the corpus were dropped by the HiFST system. This happens because numbers are often OOV: any numeral that was not present in the MT training data cannot be translated by the MT system. Numerals should nearly always be translated as themselves (besides minor edits for locale-specific punctuation). These errors can be found by a simple check for numbers that produce no target-language tokens or are translated into words that are not POS-tagged as numbers.

OOV words: If the source-language phrase is left untranslated in a different script than the target language, or if the target-language phrase is an obvious transliteration, it was probably OOV and should be flagged as an error. Note that, in Arabic, Buckwalter transliteration often produces tokens with embedded punctuation or mixed-case words, which are easy to detect.

Algorithm 3 The algorithm for detecting missing content words (or phrases). The algorithm assumes that a source-language phrase containing one or more content words should produce a target-language phrase containing at least one content word.
The function *hasContentWord* takes the POS tags and tokens from a phrase and uses language-specific (and POS tagset-specific) rules to determine whether a phrase contains a content word.
S and T represent the source- and target-language sentences, respectively.

```
function DetectMissingCWErrors(S,T)
    for all source-language phrases $S$ do
        if $hasContentWord(S)$ then
            $T \leftarrow$ target-language phrase aligned to $S$
            if $T = \emptyset$ then
                ▷ If the phrase is aligned to nothing, flag it as a deletion
                $addError(S, T)$
                continue
            end if
            if $!hasContentWord(T)$ then
                ▷ If the phrase is aligned to a phrase with no content words, flag it as a mistranslation
                $addError(S, T)$
                continue
            end if
        end if
    end for
end function
```

5.3 Correction via Re-Translation

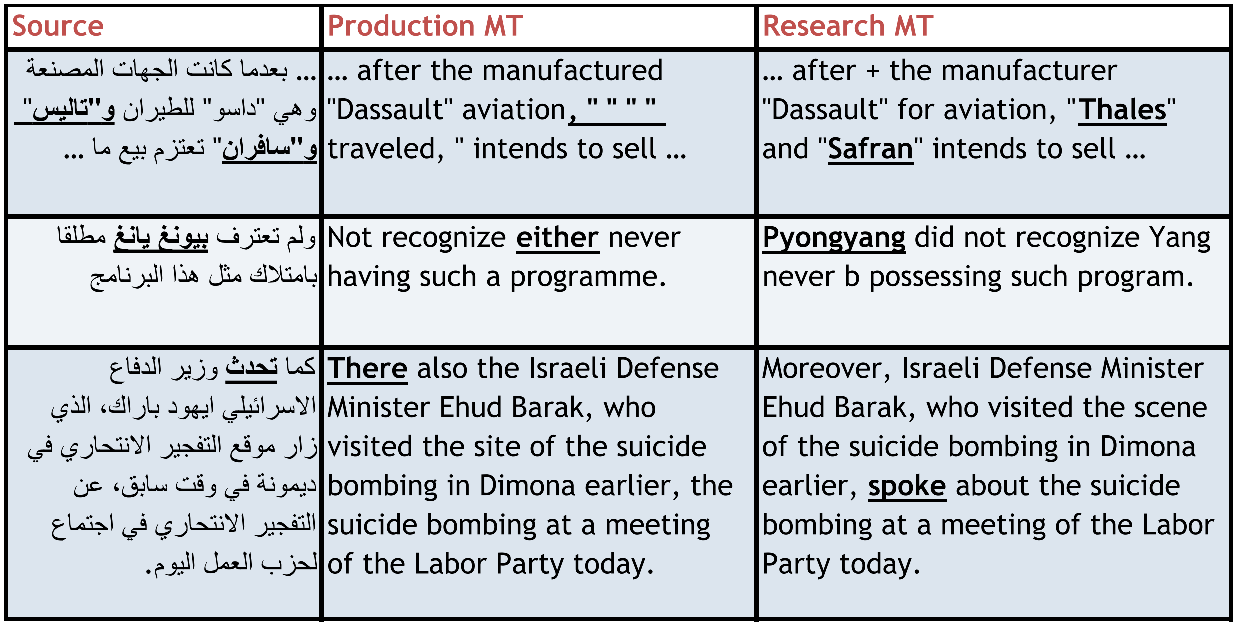
In the next chapter, we evaluate the automatic error detection algorithms as part of an automatic post-editing pipeline. But for some applications, detecting errors in MT is sufficient even without correcting the errors. If high-quality translation is crucial, flagged sentences can be passed to a human post-editor for correction or even fully re-translated by a professional translator. Alternatively, if there are many redundant sentences, flagged translations can simply be hidden: if there are 100 reviews for the same hotel, a travel site can show only the most fluently translated reviews. Finally, some applications may wish to disable MT entirely for specific contexts if they can detect that the translations are very low quality. For instance, in social media sites like Facebook and Twitter, sentences are often very difficult to translate due to dialect (in Arabic), slang, abbreviations, misspellings and non-standard typography (e.g., smiley faces or substituting numbers for words). Enabling MT only for tweets or status updates that can be translated without significant errors could prevent user frustration.

Similarly, we consider a CLQA setup where our error detection algorithms can be exploited in the absence of targeted MT error correction. First, the entire CLQA corpus is translated offline by a production MT system. A higher-quality research MT system is available but is too slow to be practical for a large amount of data. At query-time, we can call the research MT system to re-translate sentences, but due to time constraints, we can only re-translate k sentences (we set $k = 25$). The goal is to re-translate sentences with detected errors that are most likely to improve CLQA relevance. We present a heuristic for ranking translated sentences based on a relevance model and a model of error importance.

First we describe the two-stage MT for CLQA framework in more detail. Then we describe the experimental settings. Finally, we present results and discuss the lessons learned in carrying out the evaluations.

5.3.1 Two-Stage SMT for CLQA

The idea behind two-stage SMT is that the better MT system might correct adequacy errors in CLQA results before they are passed to the user. Table 5.2 shows examples of MT adequacy errors in production MT output that are detected by the error detection algorithms and subsequently



Source	Production MT	Research MT
... بعدما كانت الجهات المصنعة وهي "داسو" للطيران **و"تاليس"** **و"سافران"** تعتزم بيع ما ...	... after the manufactured "Dassault" aviation**, " " " "** traveled, " intends to sell ...	... after + the manufacturer "Dassault" for aviation, "**Thales**" and "**Safran**" intends to sell ...
ولم تعترف **بيونغ يانغ** مطلقا بامتلاك مثل هذا البرنامج	Not recognize **either** never having such a programme.	**Pyongyang** did not recognize Yang never b possessing such program.
كما **تحدث** وزير الدفاع الاسرائيلي ايهود باراك، الذي زار موقع التفجير الانتحاري في ديمونة في وقت سابق، عن التفجير الانتحاري في اجتماع لحزب العمل اليوم.	**There** also the Israeli Defense Minister Ehud Barak, who visited the site of the suicide bombing in Dimona earlier, the suicide bombing at a meeting of the Labor Party today.	Moreover, Israeli Defense Minister Ehud Barak, who visited the scene of the suicide bombing in Dimona earlier, **spoke** about the suicide bombing at a meeting of the Labor Party today.

Figure 5.2: Three examples where missing content word errors in the production MT output were fixed via re-translation by the research MT system.

fixed via re-translation by the research MT system. Due to time constraints, not all CLQA results can be re-translated, so a ranking heuristic is used to select sentences that should be re-translated.

Baseline CLQA System The baseline CLQA system translates the full corpus offline before running further processing on the translated sentences (POS tagging, parsing, NE recognition and information extraction) and indexing the corpus. At query-time, CLIR (implemented with Apache Lucene) returns documents relevant to the query, and the CLQA answer extraction system is run over the translated documents to select sentences containing answers. Since sentence selection is done completely in the target language, MT errors can significantly degrade answer extraction. Errors in MT also decrease the accuracy of target-language processors, such as NE recognition, which in turn further degrades sentence selection.

CLQA System with MT Error Detection The error detection and re-translation steps were added to the baseline system after document-level CLIR and before sentence-level answer extraction so they could have an impact on sentence selection. Error detection is done at query time so that query context can be taken into account when determining which sentences to re-translate. We also use the task context to detect errors in translating NEs present in the query.

The detailed algorithm is shown in Algorithm 4. After documents relevant to the query are retrieved by CLIR, the error detection module is run on the sentences in each document. The error detection algorithm takes as input the pseudo-parallel sentence $sent_j$, the query Q and the query translation QT, and returns the error importance score $score_{error}$. A relevance score based on bilingual bag-of-words relevance ($BOW Relevance$) is also calculated and both scores are combined into $score_{combined}$ as described below. After the error detection module finds $2k$ sentences with errors or exhausts the document list, the sentences are ranked and the top $k = 25$ sentences are re-translated. Then the merged set of original and re-translated relevant sentences are passed to the answer extraction module.

By doing re-translation prior to answer extraction, the algorithm has the potential to improve both precision and recall. We hypothesize that an improved translation of a relevant Arabic sentence is more likely to be selected by the answer extraction system (and therefore increase recall). Also, a better translation of a relevant sentence is also more likely to be perceived as relevant, thus improving precision.

Ranking Heuristics: Even if the re-translation corrects the errors in MT, if the sentences are not relevant, they will have no impact on CLQA. A bilingual bag-of-words matching model was used to rank sentences with more keyword matches to the query higher. We also heuristically ranked the types of MT errors by potential impact on the task. Errors affecting NEs (either via source-language POS tagging or source-language NE recognition) were ranked highest, since our particular CLQA task is focused on NEs. The final output of the algorithm is a list of sentences with MT errors, ranked by relevance to the query and importance of the error.

5.3.2 Experiments

Queries and Corpus: The task is a template-based CLQA task, where the questions are open-ended, non-factoid information needs. (Template-based CLQA is discussed further in section 2.2.2.) The specific question types were defined for the GALE Y4 evaluation and are listed in Appendix A. There are 22 question types, and each type has its own relevance guidelines. For instance, one question type is “Describe the election campaign of [PERSON],” and a query might ask about Barack Obama. We used 31 text queries developed by the Linguistic Data Consortium (LDC) and

39 speech queries developed by researchers at IBM. The GALE Y4 corpus included formal text (72,677 newswire documents), informal text (47,202 web documents),[1] formal speech (50 hours), and informal speech (80 hours). The speech data was story segmented and run through a speech recognition system before translation.

MT Systems: The MT systems used in this experiment were the production and research IBM DTM2 MT systems [Ittycheriah and Roukos, 2007], described in Chapter 3. DTM2 uses a maximum entropy approach to extract minimal translation blocks (one-to-M phrases with optional variable slots) and train system parameters over a large number of source- and target-language features. The research system incorporates many additional syntactic features and does a deeper (and slower) beam search, both of which cause it to be much slower than the production system. In addition, the research MT system filters the training data to match the test data, as is customary in MT evaluations, whereas the production system must be able to handle a wide range of input data. Re-training the MT system based on the test set allows more test-specific training data to be used to tailor the MT system to the input, but significantly increases the processing time.

Overall, the research MT system performs 4 BLEU points better than the production MT system on a standard MT evaluation test corpus, but at a great cost: the production MT handles 5,000 words per minute, while the research MT system handles 2 words per minute. Using 50 machines, the production MT system could translate the corpus in under 2 hours, whereas the research MT system would take 170 days. This vast difference succinctly captures the motivation behind the time-constrained re-translation step.

5.3.3 Evaluation

CLQA Evaluation: The CLQA evaluation was run on Amazon Mechanical Turk (AMT), which has been shown to have high correlation with expert annotators on many NLP tasks at a lower cost [Snow *et al.*, 2008]. It has also been used in MT evaluation [Callison-Burch, 2009], though in that evaluation, the authors used reference translations.

Annotators were first presented with template relevance guidelines and an example question, along with 3 – 4 example sentences and expected judgments. The instructions and examples are shown in appendix Figure B.1. Then a query was presented to the annotator along with

[1] The web corpus was not available at the time of our experiments, so it was not included in the evaluation.

Genre	Errors per Sentence	Errors per 1,000 Tokens
Newswire	0.16	56
Broadcast News	0.23	105
Broadcast Conversation	0.14	84

Figure 5.3: The number of automatically detected errors (of all kinds) for each genre. The broadcast conversation genre has fewer errors per sentence because it has very short sentences. When compared by errors per thousand tokens, both speech genres have significantly more detected errors than text.

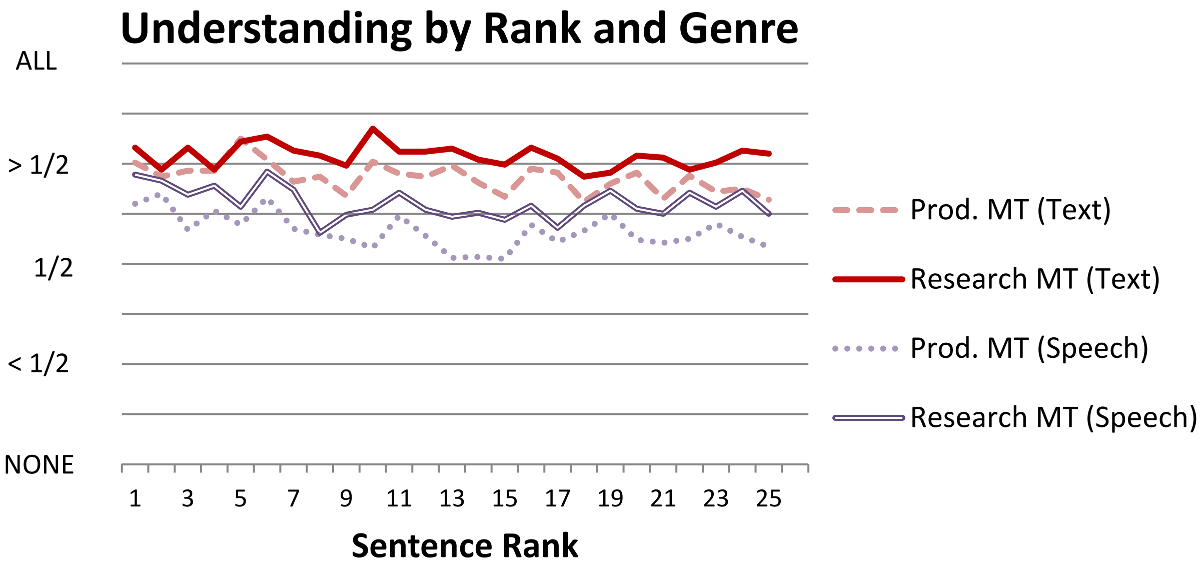


Figure 5.4: The average judged understandability by sentence rank, comparing speech and text. The y-axis is a 5-point scale from *All* to *None*, and the x-axis is the sentence rank according to the scoring heuristic. As in Table 5.3, the speech genre is much less understandable than text. Both improve significantly after re-translation, with speech gaining an average of 0.28 points and text an average of 0.22 points.

five sentences from a single MT system. For each sentence, the annotators were asked to judge understandability and relevance to the query. The judging interface is shown in appendix Figures B.2 and B.3. Each sentence was annotated in the production MT version and the research MT version.

The **understandability** rating was loosely based upon MT adequacy evaluations: annotators were told to ignore grammatical errors and focus on perceived meaning. Since we did not have reference translations, annotators were asked to rate how much of the sentence they believed they understood by selecting one of (*All*, *More than half*, *About half*, *Less than half* and *None*).

The **relevance** rating was based on the template relevance guidelines, and annotators could select one of (*Relevant*, *Maybe relevant*, *Not relevant*, *Can't tell due to bad translation* and *Can't tell due to other reason*). In practice, the latter option was rarely used (1% of ratings).

The understandability and relevance judgments are linked, since it is impossible to judge relevance on a completely incomprehensible sentence. On the other hand, sometimes relevance can be determined by a small part of the sentence. Figure B.2 shows an example annotation where the translation is so garbled that the relevance to the query cannot be determined. In the annotation in Figure B.3, the MT is also garbled, but the phrase relevant to the query is understandable, so it can be judged relevant.

For 70 queries, we evaluated the top 25 ranked sentences in both the production and research MT versions. Each sentence was judged for both relevance and understandability by five annotators each, for a total of 35,000 individual judgments. As is standard, some of the judgments were filtered due to noise by using the percent of time that an annotator disagreed with all other annotators, and the relative time spent on a given annotation. The percent of sentences with majority agreement was 91% for relevance and 72% for perceived adequacy.

Error Detection: For the intrinsic evaluation of error detection, annotators were presented with an Arabic sentence with a single token highlighted, and asked whether the token was a "content word" or not. Then annotators were asked to decide which of two translations (in random order) translated the highlighted Arabic word best, or whether they were equal. In total, 150 sentences were judged by annotators with knowledge of Arabic. For both questions, kappa agreement was moderate (0.42 for whether a token was a content word, and 0.51 for which translation was better).

5.3.4 Results

Number of Errors Detected: Table 5.3 shows how often errors were detected in different genres. Both speech genres have much higher error rates than the formal text genre. Translating speech is difficult because errors from text-to-speech cause errors in the input sentences. Furthermore, spoken language in Arabic tends to be more dialectal, which is out of domain for the MT training data. Newswire text is primarily Modern Standard Arabic, which matches the MT training data more closely.

Overall, 12% of the sentences had detected errors. After ranking and selecting the top $k = 25$ sentences, 36% of sentences with errors were re-translated.

Understandability and Relevance: Figure 5.4 compares the perceived understandability of the production MT to the re-translated research MT. The speech sentences are judged significantly less understandable than the text sentences, which mirrors the trend shown by our automatic error detectors. For both text and speech genres, the re-translated sentences are rated significantly more understandable than the production MT sentences. Re-translation has an even larger effect on speech than text: speech sentences gain an average of 0.28 points understandability, while text sentences gain 0.22 points.

Figures 5.5 and 5.6 show the effect of re-translation on understandability (across all genres) and relevance, respectively, by sentence rank. In other words, at each rank i, the understandability (or relevance) judgments from all five judges were averaged for each query-sentence pair, and then the scores from all sentences at rank i were averaged. For understandability, the research MT system consistently outperformed the production MT system by a statistically significant margin. For relevance, the research MT curve is only marginally higher than the production MT curve.

Annotators judged 14.5% of the production MT sentences relevant. After re-translation, the number of relevant sentences increased to 14.7%. The overall change was small because some sentences decreased in relevance. Table 5.2 shows the results of comparing annotations on the original MT with annotations on the re-translated MT. Relevance was classified as $\Uparrow$ or $\Downarrow$ by comparing the majority judgment of the production MT to the research MT. Understanding was classified as $\Uparrow$ or $\Downarrow$ when the difference in average rating between the two versions was greater than 1.0.

Of the sentences with better perceived MT, 7% increased in relevance, and 3% decreased in

	Relevance				
	⇑	Same	⇓	No maj./ Don't know	
MT ⇑	20	201	9	56	17%
MT same	93	919	72	212	78%
MT ⇓	2	56	4	28	5%
	7%	70%	5%	18%	

Table 5.2: Confusion matrix showing the changes in understandability (rows) versus changes in relevance (columns).

relevance. When the re-translated sentence was considered worse, there was a 2% increased in relevance and a 4% decrease. In other words, when re-translation had a positive effect, it more often led to increased relevance. However, the impact of re-translation was mixed, and none of the changes was statistically significant.

Ranking: The shape of the relevance curves shows that ranking sentences by a simple bilingual bag-of-words model was effective, since sentences that are higher ranked have higher relevance. The sentence ranks were moderately correlated with relevance judgments: the Pearson's correlation was 0.35 for the production MT, and somewhat higher, 0.38, for the research MT.

By ranking sentences with a basic relevance model, we were able to focus the scarce MT resources on sentences that are most likely to help the CLQA task. This underscores the importance of using the task context to guide MT error detection, especially in the case of time-constrained MT.

Error Detection Accuracy: While the extrinsic evaluation focused on the impact on CLQA relevance, the goal of the intrinsic evaluation was to measure the precision of the error detection algorithm, and whether re-translation with a better MT system addressed the detected errors. Of the 82% of sentences where both judges agreed, 89% of the detected errors were considered content words. All of the OOV tokens were content words (except for one disagreement). Surprisingly, for the errors involving content words, 60% of the time both MT systems were judged the same with regard to the highlighted error. The research system was better 39% of the time, and the original was better only 1% of the time (excluding 26% disagreements).

5.3.5 Discussion

The CLQA evaluation was based on three hypotheses:

- That we could detect errors in MT with high precision.
- That re-translating errorful sentences with a much better MT system would correct the errors we detected.
- That correcting errors would cause some sentences to become relevant which were not previously relevant, as in [Ma and McKeown, 2009].

The intrinsic evaluation confirmed that we can identify content word deletions in MT with high precision, thus validating the first hypothesis. However, detecting the errors and re-translating them did not lead to large improvements in CLQA relevance – the impact of increased understandability on relevance was mixed and not significant. The intrinsic evaluation explains this negative result: even though the re-translated sentences were judged significantly better, the re-translation only corrected the detected error 39% of the time. In other words, the better research MT system was making many of the same mistakes as the production MT system, despite using syntactic features and a much deeper search space during decoding. We hypothesize that the translations improved in fluency more than they improved in adequacy.

Since the second hypothesis did not hold, we need to improve our error correction algorithm before we can tell whether the third hypothesis holds. This result directly motivates the need for targeted error correction of MT.

5.3.6 Evaluating Task-Embedded MT: Lessons Learned

The challenges we faced in getting conclusive results from this experiment reflect the challenges inherent in evaluating task-based MT. In this section, we discuss the lessons learned from our evaluation and how they informed our future experiments.

Reference translations: The biggest limitation in our evaluation was a lack of reference translations. Getting annotations from bilingual annotators is expensive and time-consuming, so we relied on untrained monolingual English speakers on AMT. However, since we had no references, annotators had to estimate translation adequacy without knowing what the sentence actually meant. Though the instructions emphasized that they should judge based on meaning rather than grammaticality, in practice it is difficult to separate adequacy from fluency, in particular when there are no reference translations available for comparison. It is difficult to tell how much of the improvement in understanding is due to including more meaningful information in the translation versus fixing grammar errors; indeed, the results from the intrinsic evaluation indicate that missing word errors were often not addressed.

The fact that no reference translations were available is partly due to the separation between the fields of MT and CLIR (or CLQA). While many MT corpora have reference translations, they do not have the queries and query relevance judgments that would be necessary for a CLQA evaluation, and the corpora are typically too small to be considered realistic CLQA corpora. On the other hand, there are several large-scale CLIR (or CLQA) corpora available with relevance judgments in the source language, but they do not have reference translations available (and they are usually not machine translated either).

In this experiment, we used a corpus from a shared task that had no reference translations and no gold relevance judgments, so the evaluation was particularly difficult. In the experiments in Chapter 7, we address both of these problems by using MT corpora with reference translations to get gold relevance judgments (based on the references), but we still face the problem of having a very small evaluation corpus.

Trusted annotators: Another issue that arose in analyzing the results from the AMT evaluation was how to normalize and compare judgments by different annotators. (Since then, research has appeared that tries to address these problems [Denkowski and Lavie, 2010].) First, we had to decide which annotators to ignore because they were judging too fast or simply did not understand the task. We used standard heuristics from previous papers (e.g., amount of time per HIT), but they still involved a fair amount of hand-tuning. In later experiments, we switched to CrowdFlower (previously Dolores Labs) to filter annotators by trust. With CrowdFlower, we pre-define a set of gold questions and answers, which are then randomly mixed in with test questions for the annotators. Annotators who perform poorly on the gold questions are discarded. We found that tracking the gold questions with the lowest rate of agreement (i.e., questions that the annotators found hardest to judge correctly) also helped us refine our task instructions.

Rating versus comparing: The second annotation problem was that we were directly comparing ratings of two versions of a sentence, but each version had been judged by five different people (for a total of 10 annotators). As the WMT shared task has shown [Callison-Burch *et al.*, 2007], asking annotators to rate translations on a scale is less reliable than asking them to compare two different translations. In this experiment, asking annotators to compare two MT sentences without a ground-truth reference translation would have focused more on fluency than adequacy. In future experiments where we did have references, we carried out direct A-to-B comparisons between translated sentences, which helped the annotators focus on the differences between the sentences and led to more conclusive results.

Defining the task: Several other complications were specific to this experiment. The queries that were available for testing included some of the most difficult question templates. Some of them had multiple slots (for instance, "Describe the relationship between X and Y."), which are more complicated than single-slot templated questions. Others had very specific relevance guidelines: answers to "Find statements made by or attributed to X on Y" should include "expected statements [...] about what the person would say, is likely to say or has probably said" but should not include "opinions or beliefs of the person, if not expressed as a statement by the person."

Many of the templates had information needs that were too specific for us to concisely and precisely describe to untrained annotators. In future experiments, we tried to focus on simpler question types that had broader information needs, somewhat closer to sentence-level CLIR than CLQA (in the list of cross-lingual tasks in Chapter 2).

MT word/phrase alignments: We also found that the alignments produced by the MT system were not always accurate. This problem is not limited to the MT system from this experiment; other MT systems also produce consistently unreliable alignments. And in fact, any MT system may produce phrases using unexpected alignments, as seen in the translated phrase in Figure 5.8. In this case, the verb "said" appears before the subject "General Petraeus" in Arabic and must be moved after the subject in English. Rather than re-order the phrases, the MT systems instead uses two separate mis-translations. The first phrase, which actually means "Iraq, said General", is translated as "Iraq, General", with the verb deleted. The third phrase, which means "that the matter [quote]" is translated to "added, [quote] it". In this case, two wrongs do make a right – by doing two mis-translations, the MT system produced an adequate translation of this phrase. The incorrect translations are available because during training, the MT system learns noisy phrase translations from the parallel corpus.

Since the error detection relies heavily on the word alignments, if they are incorrect, then there will be many false positives. For later experiments, we incorporated extra steps to make sure that the missing content words were not produced by neighboring phrases.

5.4 Conclusions

We presented novel online algorithms for detecting several types of MT errors in the context of a question, and a heuristic for ranking MT errors by their potential impact on the CLQA task. While much prior work focuses on MT error analysis, there is very little prior work on detecting fine-grained MT errors without reference translations. The algorithms we described are generally applicable to any MT system that produces word or phrase alignments for its output and any language pair that can be POS-tagged. Our error detection is more fine-grained and covers more types of errors than previous work. It was able to detect errors in Arabic-English MT across multiple text and speech genres, and the intrinsic evaluation showed that the large majority of tokens flagged as errors were indeed content words.

The types of errors we focused on were motivated by the in-depth error analysis we presented in Chapter 3. Translation adequacy is essential for CLQA with result translation, but multiple types of MT errors significantly degrade adequacy. In particular, NEs are crucial in our CLQA task, but are very challenging for MT systems to translate correctly. We described two methods for detecting NE mistranslations in MT output, one of them general to any MT system and one particularly useful for CLQA-embedded MT. This query-specific NE mistranslation detection algorithm exploits the task context (the query) to flag potential MT errors in the output (the translated search results).

The other adequacy errors that we focused on were missing and deleted content words. Deleted content words are those that are simply not translated at all by the MT system, but missing content words are more subtle. We consider a source-language content word to be "missing" if it does not correspond to some content word in the translation output. This type of error may be more insidious than more overt mistranslation errors because the end user is unaware that any information was lost. For instance, if Schwarzenegger is mistranslated as $wArzngr, the user does not understand the meaning of the word, but understands that there should be information there (i.e., the word is a "known unknown"). If the same name is translated as the word "the", the translated sentence may appear perfectly fluent and adequate, without the user realizing that information was lost (i.e., the word is an "unknown unknown"). Our error detection algorithms flag both deleted and missing content words in MT output.

We experimented with using the error detection algorithms alone to improve translations for CLQA via two-stage SMT for CLQA. The large-scale CLQA evaluation confirmed that the slower research MT system was significantly better than the production MT system. Relevance judgments showed that the ranking component was crucial for directing scarce MT resources wisely, as the higher-ranked sentences were most likely to be relevant to the query, and therefore most likely to benefit the CLQA system by being re-translated.

Although the error detection algorithm correctly identified MT errors, re-translating the sentences with the errors had a negligible effect on CLQA relevance. This unexpected result may be explained by the fact that only 39% of the errors were actually corrected by the research MT system, so re-translation was not a good approach for error correction. In the next chapter, we experiment with automatic post-editing, which is a more targeted approach to correcting the MT errors detected by our algorithms.

Algorithm 4 Algorithm for integrating query-specific MT error detection into the CLQA pipeline. We attempt to correct the detected errors using re-translation with a better SMT system, but can only re-translate k sentences given the time constraints. Error detection and re-translation is done after document-level CLIR but before sentence-level CLQA, so that it can have an impact on sentence selection.

```
procedure TwoStageMTForCLQA(Q,k)
    $QT \leftarrow$ query translation of $Q$
    ▷ SMLIR returns a set of documents matching the query
    $D \leftarrow SMLIR(Q, QT)$
    for all $d \in D$ do
        for all $sent_j \in d$ do
            $score_{error} = DetectErrors(sent_j, Q, QT)$
            $score_{relevance} = BOWRelevance(sent_j, Q, QT)$
            $score_{combined} = GetRetransPriority(score_{error}, score_{relevance})$
            $list.add(sent_j, score_{combined})$
        end for
    end for
    $sort(list)$
    ▷ Re-translate the top-$k$ ranked sentences with the better MT system
    for $i = 0$ to $k$ do
        $sent_i \leftarrow list[i]$
        ▷ Replace MT and alignments with new translation by MT+
        ▷ The re-translated sentence MT+ replaces the original MT in its document
        $sent_i \leftarrow retranslate(sent_i)$
    end for
    ▷ Now $k$ of the sents in the documents $D$ have been re-translated by MT+
    ▷ Select sentences relevant to the query
    $answers \leftarrow RunAnswerExtraction(Q, D)$
    return $answers$
end procedure
```

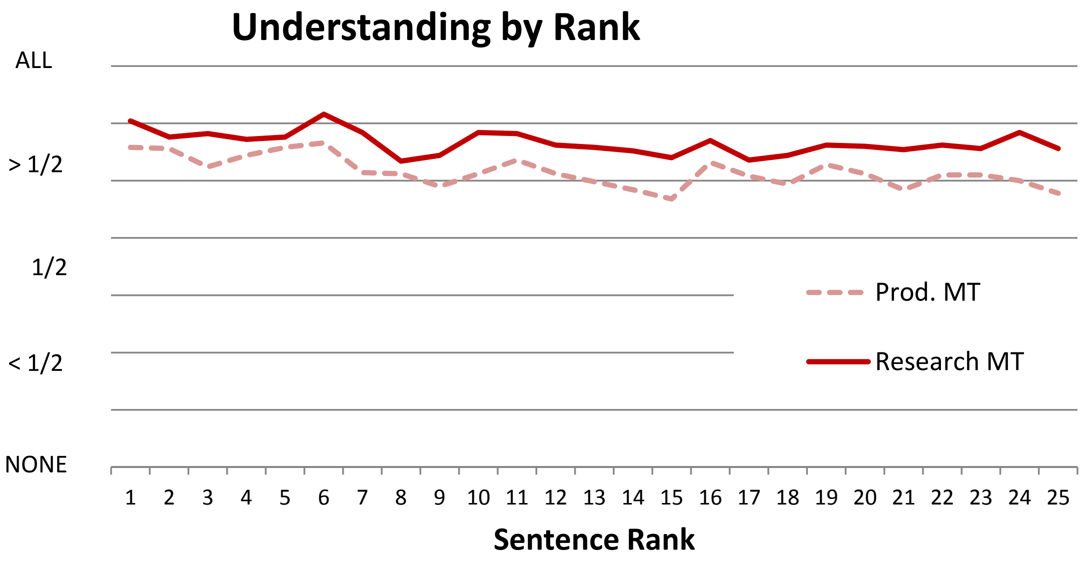


Figure 5.5: The average judged understandability by sentence rank (all genres). (The y-axis is a 5-point scale from None to All, and the x-axis is the sentence rank.)

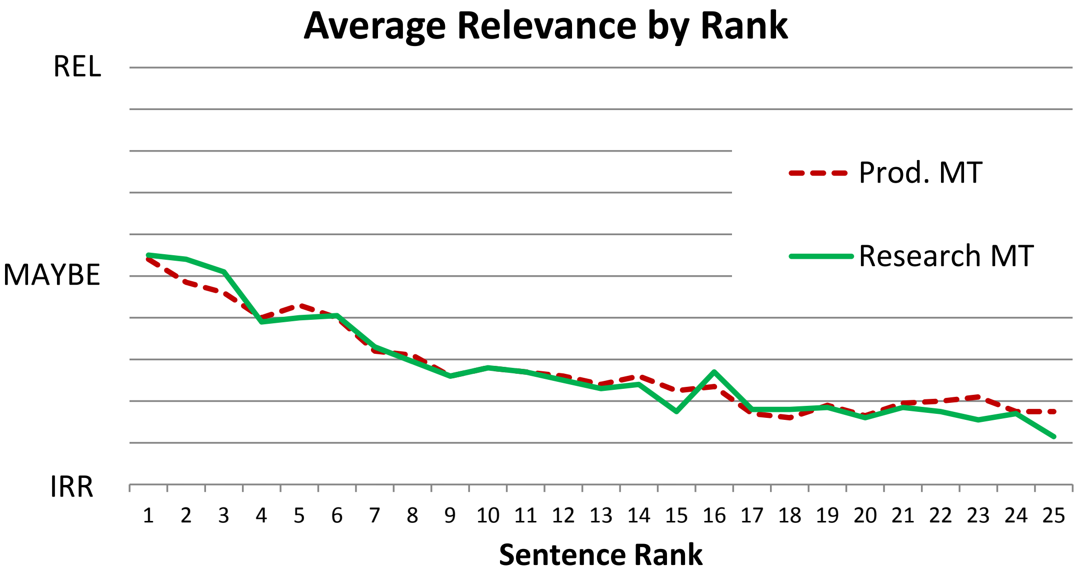


Figure 5.6: The average judged relevance by sentence rank. The y-axis is a 3-point scale: *Relevant*, *Maybe* and *Irrelevant*.

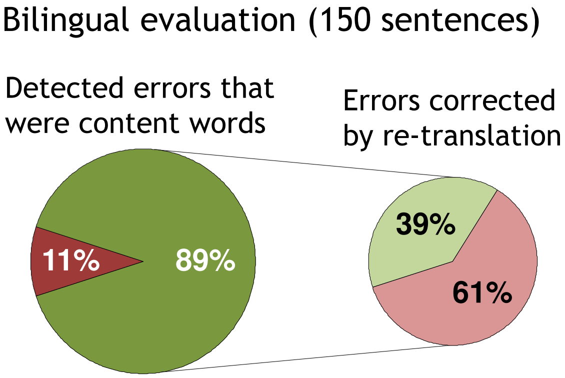
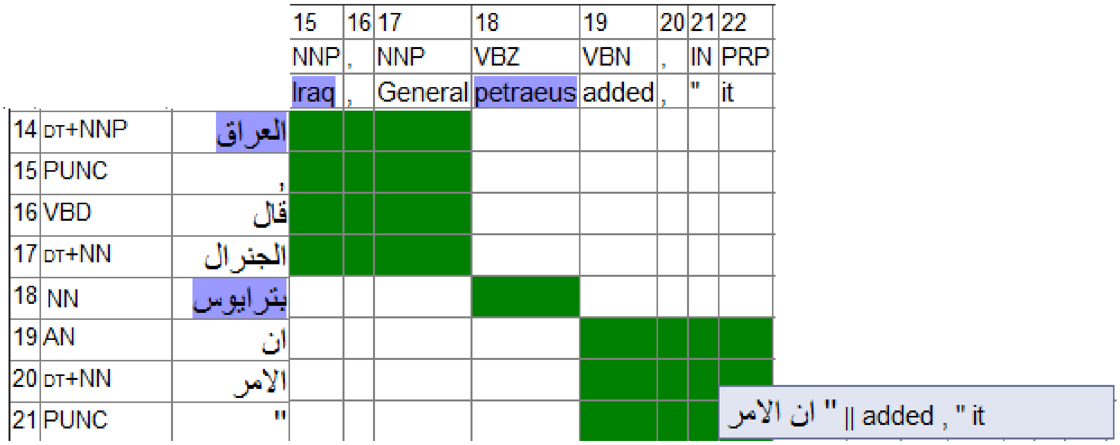


Figure 5.7: Results of the intrinsic evaluation: 89% of the detected errors were content words, but of those, only 39% of the errors were fixed by re-translation.



			15	16	17	18	19	20	21	22
			NNP	,	NNP	VBZ	VBN	,	IN	PRP
			Iraq	,	General	petraeus	added	,	"	it
14	DT+NNP	العراق	🟩	🟩	🟩					
15	PUNC	,	🟩	🟩	🟩					
16	VBD	قال	🟩	🟩	🟩					
17	DT+NN	الجنرال	🟩	🟩	🟩					
18	NN	بترايوس				🟩				
19	AN	ان					🟩	🟩	🟩	🟩
20	DT+NN	الامر					🟩	🟩	🟩	🟩
21	PUNC	"					🟩	🟩	🟩	🟩

ان الامر " || added , " it

Figure 5.8: An excerpt from the middle of a pseudo-parallel sentence, where the green blocks represent phrase alignments. The reference translation is *. . . Iraq, General Petraeus said that it "*. . . . The MT phrase is adequate, but it is produced via two mis-translations: the verb "said" is deleted in the first phrase, and the verb "added" is inserted in the third phrase.

Chapter 6

Automatically Correcting MT Errors

In the previous chapter, we presented several algorithms for automatically detecting MT errors at query-time, focusing on those that are most detrimental to CLQA with result translation. However, experiments showed that simply detecting the errors was not sufficient to improve the relevance of the results. Using a better MT system to re-translate the sentences with errors often did not fix the specific errors, because the second-pass translation was not targeted to the problem.

In this chapter, we describe several techniques to automatically correct MT errors at query time. These automatic post-editors (APEs) specifically target the fine-grained errors flagged by the error detection algorithms algorithms. We answer two of the questions posed in the last chapter: Given a phrase with a flagged error, how can we find phrase-level translations that are better than those chosen by the MT system? Once we have a list of better translations, how can we update the translated sentence to fix the error?

In order to find better phrase translations, all of the APEs use external resources as well as the task context. They each apply the corrections in different ways. The simplest techniques, the query-directed NE APE and the rule-based APE, treat MT as a black box and are applicable to any SMT system. A more sophisticated method, the direct feedback approach, treats SMT as a "glass box." In other words, the SMT system may not be altered, but the APE has some knowledge of its internal workings. This approach is general and applicable to any MT system that can accept feedback, although the details of the feedback format will be specific to a particular MT system.

In the next section, we define automatic post-editing and present related work. Then we describe a simple APE that only addresses one type of error: the query-directed NE APE. Finally, we describe our adequacy-oriented APEs, which extend the query-directed NE APE and other prior work, and we discuss the results of an in-depth human-annotated evaluation across multiple MT systems and multiple genres.

6.1 Related Work

An **automatic post-editor (APE)** is "useful for improving text produced by a wide variety of MT systems and non-native speakers" [Knight and Chander, 1994].

APEs seek to perform the same task as human post-editors: correcting errors in translations produced by MT systems. (In this thesis, when we refer to "post-editors," we always mean APEs and not human post-editors.) The advantage of post-editing is that the APE can adapt any MT output to the needs of each task without having to re-train or re-tune a specific MT system [Isabelle *et al.*, 2007]. Acquiring parallel text, training and maintaining an SMT system is time-consuming and resource-intensive, and therefore not feasible for everyone who wishes to use MT in an application. Ideally, an APE can adapt the output of a black-box MT system to the needs of a specific task in a light-weight and portable manner. Since APEs are not tied to a specific MT system, they also allow application developers flexibility in switching MT systems as better systems become available.

Adaptive APEs try to learn how to improve the translation output by adapting to the mistakes made by a specific MT system. Elming [2006] used transformation-based learning over a human post-editing training corpus to learn post-editing rules. Simard *et al.* [2007] used an SMT system to post-edit a rule-based MT system; in other words, the RBMT system translated from French into "bad English," and then the SMT system translated from "bad English" to "good English." Isabelle *et al.* [2007] further describe how to use an APE for adapting generic MT systems to specialized domains, which is closely related to our motivation in adapting a task-agnostic MT system to a specific task. Ueffing *et al.* [2008] applied this approach to Chinese-English, and extended it by annotating some target-language phrases with confidence values prior to post-editing.

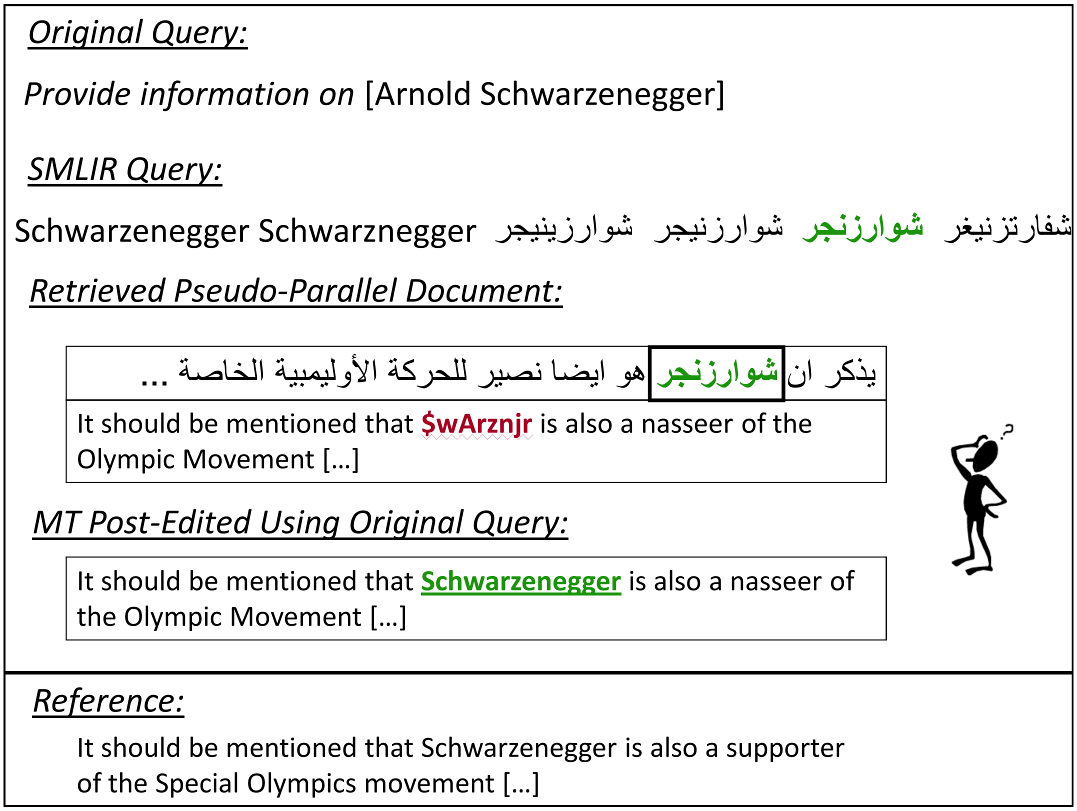
In contrast, *general* APEs can handle the translation output of any translator, from MT systems to human translators. While adaptive APEs handle many types of error from specific MT systems, general APEs target specific types of errors from any MT system. The types of errors targeted by general APEs include English determiner selection [Knight and Chander, 1994], certain types of grammar errors in English [Doyon *et al.*, 2008] and Swedish [Stymne and Ahrenberg, 2010], and complex grammatical agreement in Czech [Rosa *et al.*, 2012]. The APEs in this chapter are more similar to general APEs, since they target specific kinds of adequacy errors and are meant to work with any SMT system.

One of the general APEs is the DepFix APE, which uses a morphological tagger and a dependency parser, along with a series of post-editing rules, to correct grammatical errors in case, agreement and tense in Czech MT output. In the 2012 WMT English-Czech evaluation, the winning system was the output of an online system (anonymized as onlineB), automatically post-edited by DepFix. The onlineB+DepFix output was judged better than onlineB alone and better than 11 other English-Czech MT systems.

There are two main reasons that APEs can improve over decoder output: they can exploit information unavailable to the decoder, and they can carry out deeper text analysis that is too expensive to do in a decoder. In Ma and McKeown [2009], an APE that targets missing verbs uses the CLQA task context to select verb translations at query time. By exploiting this information, the APE was able to improve Chinese-English translation quality and the relevance of translated answers. The rule-based APE we describe extends that APE to cover additional types of adequacy errors.

Our feedback APEs are most similar to Suzuki [2011], who uses confidence estimation to select poorly translated sentences. Rather than re-translating them with feedback, translated sentences considered low-quality are passed to an adaptive SMT post-editor. This approach was successful in improving Japanese-English MT in the patent domain.

Many APEs use sentence-level analysis tools to make improvements over decoder output. Since these tools rely on having a fully resolved translation hypothesis (and since they are expensive), they are infeasible to run during decoding. The DepFix post-editor [Rosa *et al.*, 2012] parses translated sentences, and uses the bilingual parses to correct Czech morphology. While syntax-based MT systems use POS and parses, most systems do not use other types of annotations (e.g., information extraction, event detection or sentiment analysis). An alternative approach would be to incorporate these features directly into the MT system; the focus of this thesis is on adapting translations to the task without changing the MT system.



Original Query:

Provide information on [Arnold Schwarzenegger]

SMLIR Query:

Schwarzenegger Schwarznegger شفارتزنيغر شوارزنجر شوارزنيجر شوارزينيجر

Retrieved Pseudo-Parallel Document:

يذكر ان شوارزنجر هو ايضا نصير للحركة الأوليمبية الخاصة ...

It should be mentioned that **$wArznjr** is also a nasseer of the Olympic Movement […]

MT Post-Edited Using Original Query:

It should be mentioned that **Schwarzenegger** is also a nasseer of the Olympic Movement […]

Reference:

It should be mentioned that Schwarzenegger is also a supporter of the Special Olympics movement […]

Figure 6.1: Example of query-directed named entity automatic post-editing. The question is first converted into a bilingual SMLIR query. The pseudo-parallel sentence is retrieved by the Arabic part of the query. Although the sentence is relevant in Arabic, it appears irrelevant in English due to MT errors. Using word alignments and the NE from the original query, the APE rewrites the mistranslated name, and the sentence now appears relevant in translation.

6.2 Query-Directed Named Entity Automatic Post-Editor

The query-directed NE APE corrects mistranslated NEs that are present in the CLQA query. Specifically, it addresses errors detected by the query-specific NE mistranslation detection algorithm discussed in section 5.2.1. The APE uses the actual query as the improved phrase translation. It edits the sentence by simply replacing the mistranslated phrase with the NE phrase from the query.

Compared to the other APEs in this chapter, this APE is the most limited in scope, since it only addresses one type of error, yet it has the potential to greatly increase result relevance because it focuses on NEs in the query. The small-scale evaluation of this APE showed promising results and motivated us to develop the more sophisticated APEs in the next section, which we evaluate in more depth and on a larger scale.

6.2.1 Approach

This APE utilizes information from the CLIR model we introduced in Chapter 4, Simultaneous Multilingual IR (SMLIR). In SMLIR, documents are indexed as pseudo-parallel documents, containing both the source-language text and the target-language MT. Queries are also represented bilingually: they contain the original query as well as query translations. Retrieval is done simultaneously in both the source and target languages (i.e., the document and query languages).

The SMLIR approach to retrieval is illustrated in Figure 4.3. SMLIR was motivated by the error analysis in Chapter 3, which showed that MT errors led to recall errors in a purely document translation (DT) approach to CLIR. SMLIR addressed these recall errors by incorporating query translation (QT) into the CLIR model. Relevant sentences whose translations were garbled due to MT errors can still be retrieved by SMLIR, since it also retrieves in the source language. However, when garbled translations of relevant sentences are shown to the end user, they may appear irrelevant due to MT errors. The query-directed APE targets this problem.

Figure 6.1 shows the type of error addressed by the query-directed NE APE. The SMLIR query retrieves this sentence via a source-language (Arabic) match, but the document translation does not match the query due to an MT error. Since the MT system produces word alignments, after detecting an incorrect translation, the incorrect translation (“$warznjr”) can be replaced with the original query (“Schwarzenegger”). The user then gets a response with the correctly spelled name, and the translated document is perceived relevant.

The figure also illustrates the post-editing steps:

1. Use SMLIR to detect potential mistranslation: if a result contains a match in the document (source) language but not in the MT, consider it a potential error.

2. Using word alignments, extract the MT hypothesis: the query (target) language words that correspond to the source-language match.

3. If the MT hypothesis is a fuzzy match to a name variation in the dictionary, do not post-edit.

4. Use word alignments to decide which translation tokens to replace. Name translations are not necessarily contiguous, but the post-editor should not insert the name multiple times. If the match is part of a larger phrase match, there may be links to other words, and it is important not to replace other tokens in the sentence.

5. Re-write the selected tokens with the original query.

In general, a find-and-replace approach to correcting MT is too simplistic and likely to be problematic. We restrict this APE to proper names, which are more amenable to rewriting than arbitrary words or phrases. Names are particularly hard for MT systems to translate, but translating them correctly is especially important for CLIR and CLQA.

This APE has the potential to improve over the original MT decoder in two ways. First, SMT takes a sentence-by-sentence approach to translation, ignoring issues of consistency. In one document in our corpus, the same name was translated three different ways. Second, we have more information at query-time than at document translation time. In our application, we get name-tagged queries, and we can try to match them to documents that are name-tagged in two languages. Therefore, we are using information from multiple sources to make an informed translation decision. This is similar to the approach of [Ji and Grishman, 2007] for using joint inference over information extraction and entity translation to improve name translation.

6.2.2 Results

The experimental setup for evaluating this APE is the same as the one used in the SMLIR evaluation in section 4.3, except that we evaluated the APE on Arabic-English MT rather than Chinese-English

MT. The MT system was the RWTH production system, and the corpus and queries were from the GALE Y2 distillation task.

The goal of the query-directed NE APE is to use query-time information to improve the translation quality of returned results. Our evaluation indicates that, despite its simplicity, this approach is able to improve our MT output. We ran the APE on 127 Arabic GALE Y2 queries, using the top 10 document results from our SMLIR system. Of those, 28 (22%) of the queries returned documents with detected errors. Of these, 15% of the IR name matches were rewritten. For each query, up to the first 5 post-edits were examined by the author (who is a student of Arabic, but not a native speaker). The annotator decided whether the replacement was Acceptable, Not Acceptable or Ambiguous. Of the 101 rewrites examined, our replacements were Acceptable 93% of the time. 6% were Not Acceptable and 1% were Ambiguous, as shown in Figure 6.2.

Improved name translation is essential for good cross-lingual applications, since a relevant result with a poor name translation can seem irrelevant to the end-user. However, MT metrics such as the BLEU score [Papineni *et al.*, 2002b] do not take into account the relative importance of various words in the sentence. Producing an incorrect translation of a name such as "Zarqawi" has the same effect on BLEU (bilingual evaluation understudy) score as producing an incorrect determiner ("a" instead of "the"), though the latter is unlikely to diminish a reader's comprehension of the text. By using query-directed post-editing, we hope to improve result translation for CLQA.

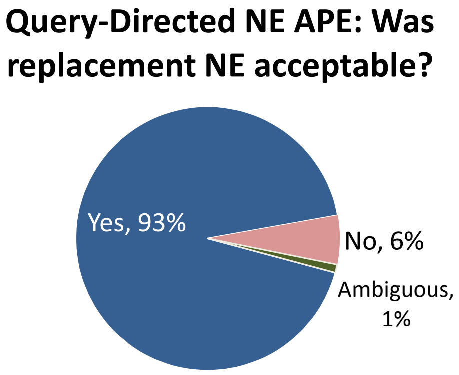


Figure 6.2: Of 101 sentences post-edited by the query-directed NE APE, 93% of the rewritten English NEs were found to be acceptable translations based on the Arabic source NE.

6.3 Adequacy-Oriented Automatic Post-Editors

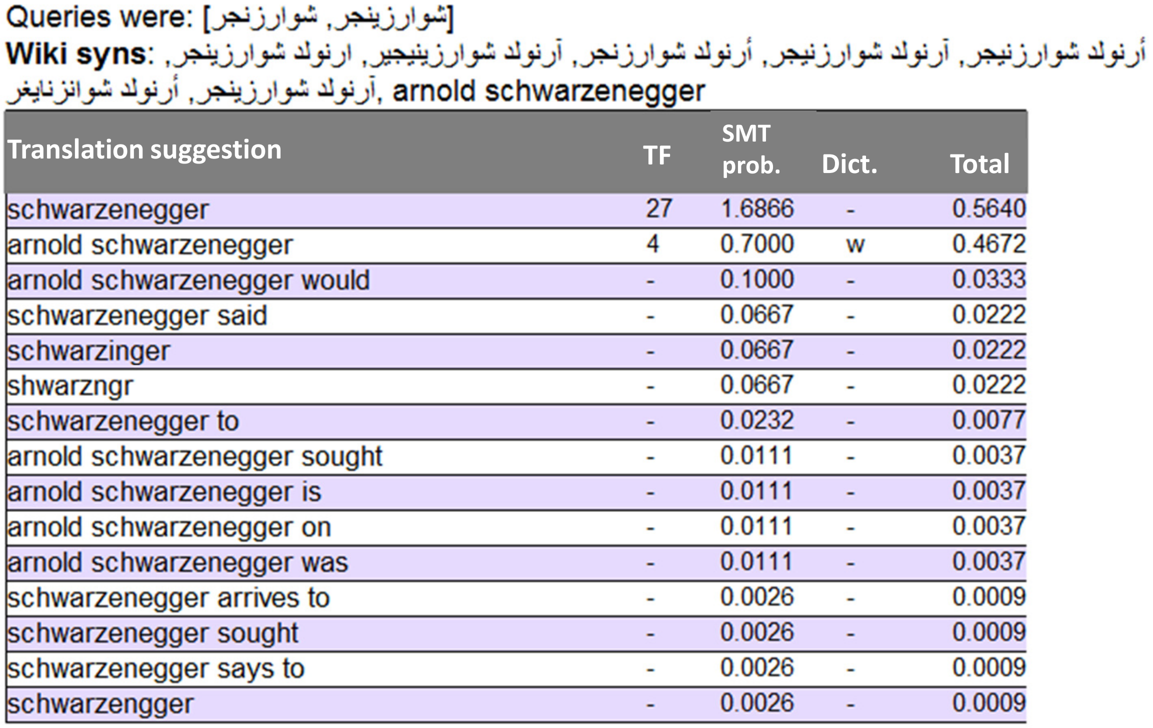
The query-directed NE APE demonstrated that even a simple APE can have a significant effect on MT adequacy. In this section, we present APEs that use more elaborate algorithms to handle a wider range of MT adequacy errors. The APEs target the adequacy errors detected by the algorithms in the previous chapter: deleted content words, content words that were translated into function words, and mistranslated NEs.

The adequacy-oriented APEs carry out three steps: 1) detect errors, 2) suggest and rank corrections for the errors, and 3) apply the suggestions. All the APEs use identical algorithms for steps 1 and 2, and only differ in how they apply the suggestions. The algorithms are language-pair independent, though we carried out all of our experiments on Arabic-English MT. The error detection step applies the algorithms described in Chapter 5, and the suggestion generation step exploits external resources and the task context to find and rank possible translations.

Once the APEs have a list of errors with possible corrections, we experiment with different ways of applying the corrections: one approach that uses phrase-level editing rules, and two techniques for passing the corrections as feedback back to the MT systems.

The *rule-based APE* uses word alignments to decide where to insert the top-ranked correction for each error into the target sentence. This approach rewrites the word or phrase where the error was detected, but does not modify the rest of the sentence. We test these MT system-independent rules on two MT systems, HiFST and Moses (described in more detail in section 6.3.4.2).

The *feedback APE* passes multiple suggestions for each correction back to the MT system, and allows the MT decoder to determine whether to correct each error and how to correct each error during re-translation. Many MT systems have a mechanism for "pre-editing," or providing certain translations in advance (e.g., for named entities and numbers). We exploit this mechanism to provide post-editor feedback to the MT systems during a second-pass translation. While post-editing via feedback is a general technique, the mechanism the decoder uses is dependent upon the implementation of each MT system: in our experiments, HiFST accepts corpus-level feedback from the APE, while Moses can handle more targeted, phrase-level feedback from the APE.



Queries were: [شوارزينجر, شوارزنجر]

Wiki syns: أرنولد شوارزنيجر, آرنولد شوارزنيجر, أرنولد شوارزنجر, آرنولد شوارزينيجير, ارنولد شوارزينجر, آرنولد شوارزينجر, أرنولد شوانزنايغر, arnold schwarzenegger

Translation suggestion	TF	SMT prob.	Dict.	Total
schwarzenegger	27	1.6866	-	0.5640
arnold schwarzenegger	4	0.7000	w	0.4672
arnold schwarzenegger would	-	0.1000	-	0.0333
schwarzenegger said	-	0.0667	-	0.0222
schwarzinger	-	0.0667	-	0.0222
shwarzngr	-	0.0667	-	0.0222
schwarzenegger to	-	0.0232	-	0.0077
arnold schwarzenegger sought	-	0.0111	-	0.0037
arnold schwarzenegger is	-	0.0111	-	0.0037
arnold schwarzenegger on	-	0.0111	-	0.0037
arnold schwarzenegger was	-	0.0111	-	0.0037
schwarzenegger arrives to	-	0.0026	-	0.0009
schwarzenegger sought	-	0.0026	-	0.0009
schwarzenegger says to	-	0.0026	-	0.0009
schwarzengger	-	0.0026	-	0.0009

Figure 6.3: The suggestions generated for a query containing two different Arabic spellings of "Schwarzenegger" along with their weights. The term frequency (TF) comes from the indexed pseudo-parallel corpus, the phrase table probability (SMT prob.) comes from the Moses phrase table (it may be > 1 because it combines the entries from both Arabic query terms), and the "Dict." matches come from the NE dictionary ("w" means it matched). The Total column shows the combined translation suggestion confidence. In this case, the top suggestion is correct. Other suggestions arise from bad alignments during MT training ("schwarzenegger said") as well as misspellings in the parallel corpus training data ("schwarzinger").

6.3.1 Suggesting Corrections

Recall that the error detection algorithms flag specific types of phrase-level errors in MT. Each flagged error consists of one or more source-language tokens and zero or more target-language tokens. In the error correction step, the source and target sentences and all the flagged errors are passed to the suggestion generator, which uses the following three resources.

Phrase Table: The phrase table from Moses is used as a phrase dictionary (described in more detail in 6.3.4.2). The translation probabilities are used as suggestion confidences. (Using this alone with the Moses system would generate the same suggestions already considered by the decoder, but combining the phrase table probabilities with confidence levels from the other resources yields a different distribution over possible translations.)

Dictionaries: We also use the high-precision NE dictionary we described in Chapter 4. This includes a translation dictionary extracted from Wikipedia, a bilingual name dictionary extracted from the Buckwalter analyzer [Buckwalter, 2004] and an English synonym dictionary from the CIA World Factbook.[1] They are high precision and low recall: most errors do not have matches in the dictionaries, but when they do, they are often correct, particularly for NEs.

Background MT corpus: Since our motivation is CLQA, we also draw on a resource specific to CLQA: a background corpus (the GALE Y4 corpus) of about 120,000 Arabic newswire and web documents that have been translated into English by the IBM DTM2 production system. Ma and McKeown [2009] were able to exploit a similar pseudo-parallel corpus to correct deleted verbs, since words deleted in one sentence are frequently correctly translated in other sentences. Suggestion confidence is based on normalized term frequency in the indexed corpus. Using this pseudo-parallel data gives us noisy translations with perfect alignments, rather than the perfect translations with noisy alignments typically extracted from true parallel corpora.

Each of these resources is indexed using Apache Lucene[2]. For each error, the source-language phrase is converted into a query to search all three resources. Then the target-language results are aggregated and ranked by overall confidence scores. The confidence scores are a weighted combination of phrase translation probability, number of dictionary matches and term frequencies in the background corpus. The weights were set empirically on a development corpus since we had

[1] http://www.cia.gov/library/publications/the-world-factbook

[2] http://lucene.apache.org

no training data available to tune the weights automatically.

Specifically, the scoring equation was:

$$(1 + ind_{dict}) * (\frac{1}{n}\sum_{1}^{n} weight_i * score_i)$$

where: n is the number of suggestion look-up resources that return scored suggestions; $score_i$ is the normalized score from one suggestion resource; $weight_i$ is the weight assigned to resource i and ind_{dict} is an indicator function that returns 1.0 when the suggestion is found in a dictionary or 0.0 otherwise. If the suggestion was found in the dictionary but no other resource, the score was given a constant value (of 0.5). A ceiling of 1.0 was applied so that scores fell between 0 and 1.

In our Arabic-English experiment, we used the high-precision NE dictionary for the indicator function as well as scores from three resources: the phrase table probability, a normalized term frequency over the entire background corpus and a normalized term frequency over related documents/sentences. We used uniform weights for these three resources, though if one had more confidence in some resources, it would make sense to weight scores from those resources higher. In future work, it would be better to tune the parameters automatically rather than combining the scores in this ad hoc manner, but that would first involve additional data collection.

Figure 6.3 shows the output of the suggestion generator for a SMLIR query containing two different spellings of Schwarzenegger. The query was further expanded using synonyms from Wikipedia, shown as "wiki syns." The leftmost column shows the ranked translations, and the other columns show the score from each resource. Finally, the rightmost column ("Total") shows the combined scores.

6.3.2 Rule-Based APE

Table 6.1 shows examples of sentences post-edited by the different APEs. For each error, the rule-based post-editor applies the top-ranked correction using one of two operations: *replace* or *insert*. An error can be replaced if there is an existing translation, and all of the source- and target-language tokens aligned to the error are flagged as errors. (This is to avoid over-writing a correct partial phrase translation, as in the second example where the word "their" is not replaced.) If the error cannot be replaced, the new correction is inserted.

During the *replace* operation, all the original target tokens are deleted, and the correction is inserted at the index of the first target token.

For the *insert* operation, the algorithm first chooses an insertion index, and then inserts the correction. The insertion index is chosen based on the indices of the target tokens in the error. If there are no target tokens, the insertion index is determined by the alignments of the neighboring source tokens. If they are aligned to neighboring translations, the correction is inserted between them. Or, if only one of them is aligned to a translation, the correction is inserted adjacent to it. If an insertion index cannot be determined via rules, the error is not corrected.

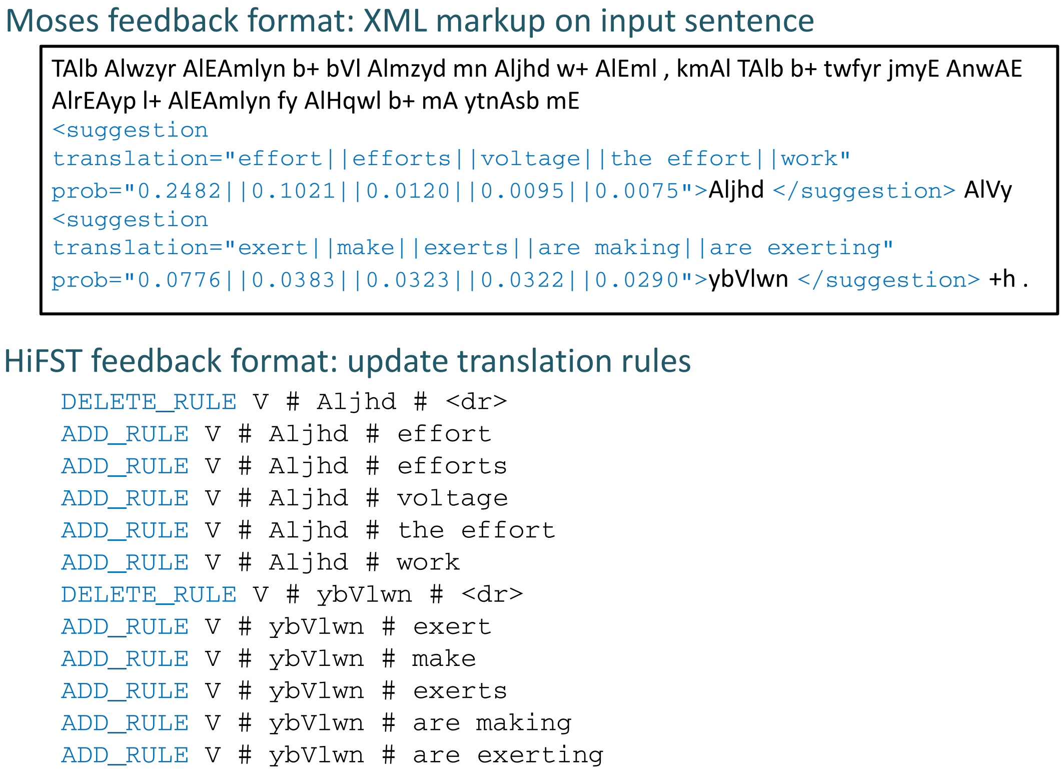
These editing rules are MT system-independent, language-independent and relatively simple. The word order is copied from the original translation or from the source sentence. This simple model worked for the query-directed APE because it was rewriting mistranslated NEs that were already present in the translation. Similarly, Ma and McKeown [2009] successfully re-inserted deleted verbs into English translations using only word alignments, assuming that local Chinese SVO word order would linearly map to English word order.

However, the adequacy-oriented APEs need to deal with a much wider range of error types, including phrases that were mistranslated, partially translated or never translated; and content words of any POS, not just NEs or verbs. Since Arabic word order differs from English, these rules often produce poorly ordered words: verbs may appear before their subjects, and adjectives may appear after their nouns. In this case, we are explicitly trading off fluency for adequacy, under the assumption that the end task is adequacy-oriented. In the first sentence in Table 6.1, the subject comes after the auxiliary verb, but the sentence can still be understood. On the other hand, since adequacy and fluency are not independent, degrading the fluency of a sentence can often negatively impact the adequacy as well.

Even when the error detection and correction steps work correctly, not all errors can be fixed with these simple operations. The original MT may be too garbled to correct, or may have no place to insert the corrected translation so that it carries the appropriate meaning.

6.3.3 Feedback APEs

To mitigate the problems of the rule-based APE, we developed an approach that is more powerful and flexible. The feedback APEs take as input the same list of errors and corrections as the rule-based APE, and then convert the corrections into feedback for the MT system. Sentences with



Moses feedback format: XML markup on input sentence

```
TAlb Alwzyr AlEAmlyn b+ bVl Almzyd mn Aljhd w+ AlEml , kmAl TAlb b+ twfyr jmyE AnwAE
AlrEAyp l+ AlEAmlyn fy AlHqwl b+ mA ytnAsb mE
<suggestion
translation="effort||efforts||voltage||the effort||work"
prob="0.2482||0.1021||0.0120||0.0095||0.0075">Aljhd </suggestion> AlVy
<suggestion
translation="exert||make||exerts||are making||are exerting"
prob="0.0776||0.0383||0.0323||0.0322||0.0290">ybVlwn </suggestion> +h .
```

HiFST feedback format: update translation rules

```
DELETE_RULE V # Aljhd # <dr>
ADD_RULE V # Aljhd # effort
ADD_RULE V # Aljhd # efforts
ADD_RULE V # Aljhd # voltage
ADD_RULE V # Aljhd # the effort
ADD_RULE V # Aljhd # work
DELETE_RULE V # ybVlwn # <dr>
ADD_RULE V # ybVlwn # exert
ADD_RULE V # ybVlwn # make
ADD_RULE V # ybVlwn # exerts
ADD_RULE V # ybVlwn # are making
ADD_RULE V # ybVlwn # are exerting
```

Figure 6.4: The feedback APE format for Moses and HiFST. For Moses, the input source (in safe Buckwalter transliteration) is augmented with XML markup containing the feedback, including translation probabilities. For HiFST, the feedback is passed as updates to the global translation rule table: new rules (also in safe Buckwalter) may be added, and existing rules may be deleted.

detected errors are decoded a second time with feedback. Passing feedback to the MT system is a general technique: many MT systems allow users to specify certain fixed translations ahead of time, such as numbers, dates and named entities. The underlying implementation of how these fixed translations are handled by the decoder is MT system-specific, and we describe two such implementations in section 6.3.3.1: corpus-level feedback and phrase-level feedback.

The difference between pre-editing and post-editing in this case is that the post-editor is *reactive* to the first-pass translation. The APE only passes suggestions to the MT system when it detects an error in the first-pass translation, and has some confidence that it can provide a reasonable correction. Since the post-editing is actually done by the decoder, the effectiveness of the feedback

APE will vary across different MT systems.

This is an extension of the two-pass MT system from the previous chapter, where sentences with detected errors were re-translated using a much better (but slower) MT system. In that experiment, we found that the second-pass translations were much better than the first-pass translations, but most of the detected errors were still present. The feedback post-editor allows us to pass specific information about which errors to correct and how to correct them to the original MT system. Unlike adaptive post-editors, where the second translation step translates from "bad" target-language text to "good" target-language text, the feedback APEs re-translate from the source text, and only one MT system is needed.

The biggest advantage the feedback APEs have over the rule-based APE is that the MT system can modify the whole sentence during re-translation, while taking the feedback into account, rather than just replacing or inserting a single phrase at a time. The decoder will not permit local disfluencies that might occur from a simple insertion (e.g., "they goes" or "a impact"), and will often prefer the correct word order, as in the first sentence in in Table 6.1.

Furthermore, the decoder can take all of the feedback into account at once, whereas the rule-based approach makes each correction in the sentence separately, as in the second sentence in Table 6.1. Finally, the rule-based approach always picks the top-ranked correction for each error, and almost always edits every error. The feedback APEs can pass multiple corrections to the MT system, often along with probabilities, which proves helpful in the fourth sentence in Table 6.1. One drawback of the feedback APEs is that they are slower than the rule-based APE since they require a second-pass decoding. Also, the decoder may ultimately decide not to use any of the corrections, which may be an advantage if low-confidence suggestions are discarded, or could be a disadvantage, since fewer errors will get corrected.

6.3.3.1 Corpus-Level vs. Phrase-Level Feedback

Each of our MT systems has a different mechanisms for accepting feedback on-the-fly, and handles the feedback differently. Examples of the two different feedback formats are shown in Figure 6.4. Moses allows *phrase-level feedback* with translation probabilities. Each source phrase flagged as an error is annotated with the list of possible corrections and their translation probabilities. HiFST allows *corpus-level feedback* without translation probabilities. In other words, the APE passes all of the translation suggestions for the entire corpus back to the MT system during re-translation.

Both MT systems allow multiple corrections for each detected error, unlike the rule-based APE. Both also allow the post-edited corrections to compete with existing translations in the system, so the re-translation may not use the suggested translations. Note that both forms of feedback are used in an online manner by the SMT systems; no re-training or re-tuning is done.

Overall, the phrase-level Moses feedback mechanism is more fine-grained because corrections are targeted at specific errors. On the other hand, the coarser, corpus-level HiFST feedback could result in unexpected improvements in sentences where errors were not detected, since the translation corrections can be used in any re-translated sentence.

6.3.4 Experiments

We tested our APEs on two different MT systems using the NIST MT08 newswire (nw) and web (wb) testsets, which had 813 and 547 sentences, respectively. The translations were evaluated with multiple automatic metrics as well as crowd-sourced human adequacy judgments.

6.3.4.1 Pre-Processing

The Arabic source text was analyzed and tokenized using MADA+TOKAN [Habash *et al.*, 2009]. Each MT system used a different tokenization scheme, so the source sentences were processed in two separate pipelines. Separate named entity recognizers (NER) were built for each pipeline using the Stanford NER toolkit [Finkel *et al.*, 2005], by training on CoNLL and ACE data. Each translated English sentence was re-cased using Moses and then analyzed using the Stanford CoreNLP pipeline to get POS tags [Toutanova *et al.*, 2003] and NER [Finkel *et al.*, 2005].

6.3.4.2 MT Systems

We used state-of-the art Arabic-English MT systems with widely different implementations. HiFST was built using HiFST [de Gispert *et al.*, 2010], a hierarchical phrase-based SMT system implemented using finite state transducers. It is trained on all the parallel corpora in the NIST MT08 Arabic Constrained Data track (5.9M parallel sentences, 150M words per language). It uses a two pass decoding process. The first-pass 4-gram language model (LM) is trained on the English side of the parallel text and a subset of Gigaword 3. The second-pass 5-gram LM is a zero-cutoff stupid-backoff [Brants *et al.*, 2007] estimated using 6.6B words of English newswire text.

Moses was built using Moses [Koehn *et al.*, 2007], and is a non-hierarchical phrase-based system. It is trained on 3.2M sentences of parallel text (65M words on the English side) using several LDC corpora including some available only through the GALE program (e.g., LDC2004T17, LDC2004E72, LDC2005E46 and LDC2004T18). The data includes some sentences from the ISI corpus (LDC2007T08) and UN corpus (LDC2004E13) selected to specifically add vocabulary absent in the other resources. The Arabic text is tokenized and lemmatized using the MADA+TOKAN system [Habash *et al.*, 2009]. Lemmas are used for Giza++ alignment only. The tokenization scheme used is the Penn Arabic Treebank scheme [Habash, 2010; Sadat and Habash, 2006]. The system uses a 5-gram LM that was trained on Gigaword 4. Both systems are tuned for BLEU score using MERT.

Both MT systems have mechanisms for explicitly dropping words: the Moses system deletes OOV words by default, and the HiFST system has both "drop rules" and "OOV rules" that result in words or phrases being deleted by the system.[3] Since our evaluation is focused on adequacy, we also experimented with versions of each of these systems that drop words less frequently. The **moses-keep** system retains OOV words in the MT output and transliterates them using Buckwalter. The **HiFST-noDrop** system is not allowed to use drop rules, and was re-tuned for BLEU score without drop rules. (Note that HiFST-noDrop still deletes OOV words.) We refer to these systems as **lessDrop** versions of the MT systems. We present results for them alongside results for Moses and HiFST, and discuss them later in section 6.4.

6.3.4.3 Feedback APE Implementations

The Moses MT system has a built-in feature for accepting additional phrase translations at decoding time, while the HiFST system did not originally have this feature. To implement corpus-level feedback in HiFST, we reformulate the APE translation suggestions as updates to existing HiFST translation rules. Then we apply the APE updates to the global rule table and re-decode the

[3] Note that this is a different usage of the term "rule" than above. In HiFST, a "rule" corresponds to a phrase translation, which resembles a rule in a grammar because it may contain non-terminals. In our rule-based APE, a "rule" is a deterministic method for replacing or inserting a token into an existing translation.

sentences flagged with errors. In our experiments, the top-5 feedback rules were added with uniform high probabilities, so while they do compete with existing translation rules, they have a much higher prior. Figure 6.4 shows the HiFST feedback format.

The feedback format in Moses is XML-based: the decoder accepts input sentences where words or phrases can be tagged with translations and optional probabilities, as shown in Figure 6.4. The targeting is very specific: if a word is translated correctly in the beginning of a sentence and incorrectly later in the sentence, the feedback format allows the APE to correct the second instance of the word only. At decoding time, the translations can override the phrase table (the "exclusive" option) or can compete with existing phrase table translations ("inclusive"). In our experiments, we used the "inclusive" option with the top-5 translation suggestions and include the suggestion scores as translation probabilities.

6.3.4.4 Automatic and Human Evaluation

We ran several automatic metrics on the baseline MT output and the post-edited MT output: BLEU [Papineni *et al.*, 2002b], Meteor-a [Denkowski and Lavie, 2011] and TERp-a [Snover *et al.*, 2009]. BLEU is based on n-gram precision, while Meteor takes both precision and recall into account. TERp also implicitly takes precision and recall into account, since it is similar to edit distance. Both Meteor and TERp allow more flexible n-gram matching than BLEU, since they allow matching across stems, synonyms and paraphrases. Meteor-a and TERp-a are both tuned to have high correlation with human adequacy judgments.

In contrast to automatic system-level metrics, human judgments can give a nuanced sentence-level view of particular aspects of the MT. In order to compare adequacy across APEs, we used human annotations crowd-sourced from CrowdFlower.[4] We ran a number of pilot studies to precisely design the judgment interface and instructions. Since our annotators are not MT experts, we used a head-to-head comparison rather than a 5-point scale. Adequacy scales have been shown to have low inter-annotator agreement [Callison-Burch *et al.*, 2007]. Each annotator was asked to select which of two sentences matched the meaning of one reference sentence the best, or to select "about the same." The tokens that differed between the translations were automatically highlighted, and their order was randomized. The instructions explicitly said to ignore minor

[4] http://www.crowdflower.com

grammatical errors and focus only on how the meaning of each translation matched the reference, and included a number of example judgments. The adequacy annotation instructions are in appendix Figure B.4, and two example adequacy annotations are shown in appendix Figures B.5 and B.6.

We compared each post-edited sentence to the baseline MT. For each comparison, we collected five "trusted" judgments (as defined by CrowdFlower) according to how well they did on our gold-standard questions. For clarity, we are reporting results using macro aggregation, in other words, the number of times overall that a particular APE was voted better than, worse than, or about the same as the original MT.

6.3.5 Results

Table 6.2 shows the percentage of sentences with detected errors for which the correction algorithm found a suggested translation. These sentences were passed to each APE, which could then decide to modify the sentence or leave it unchanged. The percentage of all sentences that were changed by each APE is also shown in Table 6.2.

The web genre has more errors than the newswire genre, likely because informal text is more difficult for both MT systems to translate. HiFST has twice as many sentences with detected errors as Moses. This is not a reflection of relative MT quality (both systems have comparable BLEU scores), but rather a limitation of the error detecting algorithm. When HiFST deletes a word, it is frequently dropped as a single token, which is simple to detect as a null alignment. Missing words in Moses are frequently deleted as part of a phrase, so they are more difficult to detect (e.g., mistranslating "white house" as "white" does not get flagged).

The impact of the APEs also varies depending on how many sentences with detected errors were actually changed by the APE. The rule-based APE almost always applies the edits. The corpus-level APE also modified most of the sentences, since all of the corrections were applied to all of the re-translated sentences. However, the phrase-level feedback APE frequently retained the original translation.

Both of these factors mean that the potential improvement from post-editing varies significantly by experimental setting, from only 15% of the sentences by the phrase-based feedback (Moses) on the news corpus, up to 64% of the corpus by the rule-based APE for HiFST on the web corpus.

<table>
<tr><th>Post-Editor</th><th>Sentence</th></tr>
<tr><td>Source</td><td>قد افرج عن فعنونو في ابريل (نيسان) ٢٠٠٤...</td></tr>
<tr><td>Ref.</td><td>Vanunu was released in April, 2004 ...</td></tr>
<tr><td>HiFST orig.</td><td>And was released in April, 2004 ...</td></tr>
<tr><td>Rule-Based</td><td>And was vanunu released in April, 2004 ...</td></tr>
<tr><td>Corpus-Level</td><td>Vanunu was released in April, 2004 ...</td></tr>
<tr><td colspan="2">Both post-editors re-insert the deleted name, but the rule-based version has poor word order.</td></tr>
<tr><td>Source</td><td>في الحقول بما يتناسب مع الجهد الذي يبذلونه.</td></tr>
<tr><td>Ref.</td><td>...in proportion to the efforts they make.</td></tr>
<tr><td>Moses orig.</td><td>...commensurate with their.</td></tr>
<tr><td>Rule-Based</td><td>...commensurate with effort exert their.</td></tr>
<tr><td>Phrase-Level</td><td>...commensurate with the work they do.</td></tr>
<tr><td colspan="2">The rule-based PE makes two separate edits to re-insert efforts and make, and ends up garbled. The feedback PE takes both into account at once, and comes out idiomatic.</td></tr>
<tr><td>Source</td><td>لماذا تتبرع ارامكو بمبلغ ٨ الالاف دولار...</td></tr>
<tr><td>Ref.</td><td>Why does Aramco donate 8 thousand dollars ...</td></tr>
<tr><td>HiFST orig.</td><td>Why ARAMCO to $ thousands ...</td></tr>
<tr><td>Rule-Based</td><td>He donates why ARAMCO the amount of dollars to $ thousands ...</td></tr>
<tr><td>Corpus-Level</td><td>Why Aramco donate $ 8 of thousands of dollars ...</td></tr>
<tr><td colspan="2">Both PEs re-insert the deleted verb, but the phrase-level is better. $ is incorrectly detected as a function word, and both PEs incorrectly re-insert “dollars”. The phrase-level PE avoids adding the redundant “the amount of”. <i>(The Arabic grammatical error (الالاف) was present in the input sentence, likely because this is from the web; it does not affect post-editing.)</i></td></tr>
<tr><td>Source</td><td>...وزارة الداخلية بدات تحديد صلاحيات الهيئة!!</td></tr>
<tr><td>Ref.</td><td>...Ministry of Interior Starts to Define Committee’s Authority!!</td></tr>
<tr><td>Moses orig.</td><td>...The Ministry of Interior started to define the terms of the !</td></tr>
<tr><td>Rule-Based</td><td>...The Ministry of Interior started to define the terms of body !</td></tr>
<tr><td>Phrase-Level</td><td>...The Interior Ministry started the authority of the board !</td></tr>
<tr><td colspan="2">The original sentence deletes the noun Committee. The rule-based version has the wrong translation and is ungrammatical. The phrase-level feedback selects a better translation, but the verb (define) is now deleted.</td></tr>
</table>

Table 6.1: Examples of the kinds of edits (both good and bad) made by different APEs.

MT	set	APE	sentences w/error	sentences modified
HiFST	nw	rule-based	48%	41%
		corpus-level feedback	48%	40%
	wb	rule-based	69%	64%
		corpus-level feedback	69%	62%
Moses	nw	rule-based	24%	24%
		phrase-level feedback	24%	15%
	wb	rule-based	34%	34%
		phrase-level feedback	34%	25%
HiFST-noDrop	nw	rule-based	27%	24%
		corpus-level feedback	27%	15%
	wb	rule-based	38%	37%
		corpus-level feedback	38%	25%
Moses-keep	nw	rule-based	23%	23%
		phrase-level feedback	23%	13%
	wb	rule-based	31%	31%
		phrase-level feedback	31%	22%

Table 6.2: The percentage of all sentences with errors detected, and the percentage of all sentences modified by each APE. The number of sentences is 813 for nw and 547 for wb.

6.3.5.1 Automatic Metric Results

Figure 6.3 shows all automatic metric scores for all MT systems across both genres. The exact scores are shown in Table 6.3, which shows the raw score for the baseline MT, followed by the change in score between the post-edited MT and the baseline MT. Since post-editing only changes a fraction of sentences in the corpus, the score changes are generally small.

All APEs improve the TERp-a score across all conditions[5], with the feedback APEs often outperforming the rule-based APE. The feedback APEs also improve the Meteor-a score across all

[5] Since TERp is an error metric, smaller scores are better.

		Δ BLEU			Δ TERp-adeq			Δ Meteor-adeq		
		base	rule	feed	base	rule	feed	base	rule	feed
MT	set	MT	based	back	MT	based	back	MT	based	back
HiFST	nw	51.32	−0.91	−0.41	37.49	−0.54	−0.74	69.48	+0.15	+0.32
	wb	36.15	−1.41	+0.03	60.66	−1.34	−2.69	55.24	+0.15	+0.88
Moses	nw	51.23	−0.49	+0.05	35.31	−0.22	−0.26	70.38	+0.00	+0.17
	wb	37.60	−0.50	−0.12	55.97	−0.26	−0.23	57.06	−0.07	+0.13

Table 6.3: The effect of APEs on automatic metric scores. Base columns show the score for the original MT and the other columns show the difference between the post-edited MT and the original MT. The rule-based APE is the same for both systems, and the feedback APE is corpus-level for MT A and phrase-level for MT B.

conditions, while the rule-based APE has mixed Meteor results. None of the APEs improve the BLEU score: the rule-based APE is always significantly worse than the original MT, while the feedback APEs have either a negative or negligible impact.

The positive improvements in TERp-a and Meteor-a suggest that the APEs are improving adequacy. In general, the feedback APEs improve the automatic scores more than the rule-based APE, although the rule-based APE actually edits more sentences in the corpus than the feedback APEs. The feedback APEs also always have better BLEU scores than the rule-based APE. The negative impact of APEs on BLEU score is not surprising, since they work by adding content to the translations, which is more likely to improve translation recall than precision.

6.3.5.2 Human-Annotated Adequacy Results

Figure 6.6 shows the percentage of post-edited sentences that were judged more adequate, less adequate or the same as the original MT, and the percentage of sentences with errors that the APE did not edit. For instance, for the HiFST MT system in the newswire genre (the leftmost columns on the top chart), the rule-based APE was found more adequate than the baseline MT 50% of the time; less adequate 17% of the time; about the same 20% of the time; and chose not to edit the sentence 13% of the time.

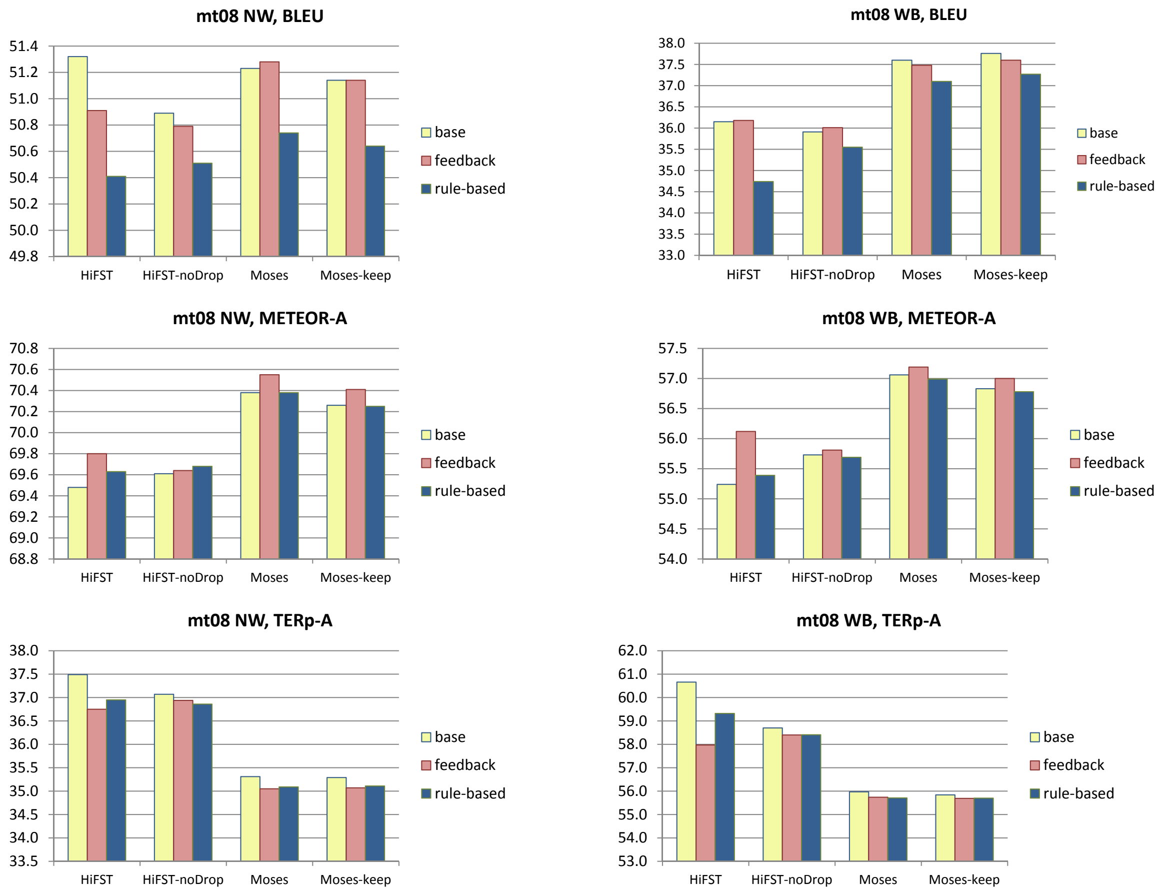


Figure 6.5: Automatic metric scores for the baseline MT and both APEs across all test sets and MT systems (including the lessDrop variants, discussed in section 6.4). Note that TERp-A is an error metric, so smaller scores are better. While the APEs usually decrease BLEU score, they always improve TERp-A. The feedback APE always improves Meteor-A also; Meteor-A scores for the rule-based APE are mixed.

Of the sentences that were post-edited, the APEs improved adequacy 30-56% of the time. Across both MT systems and both datasets, post-editing improved adequacy much more often than it degraded it: the ratio of improved sentences to degraded sentences varied from 1.7 to 4.1. For both MT systems, the APEs had a larger impact on the web corpus than the newswire corpus, both because more errors were detected in the web corpus and because the APEs edited errors more often in the web corpus.

We were surprised to find that the rule-based APE improved adequacy more often than the feedback APEs, across both MT systems and genres, especially given that the automatic metrics favored the feedback APEs. To understand the results better, we did another crowd-sourced evaluation, comparing the fluency of the rule-based and feedback post-edited sentences (when both APEs made changes). The instructions for the fluency evaluation are in appendix Figure B.7, and two example fluency annotations are shown in appendix Figures B.8 and B.9.

Results of the fluency evaluation are shown in Figure 6.7. The sentences produced by the feedback APEs were judged more fluent than the rule-based APE sentences across all conditions. The feedback post-editor was preferred twice as often (or more) as the rule-based post-editor.

The fluency evaluation shows the relative advantages of the different approaches. The rule-based APE does introduce new, correct information into the translations, but at the expense of fluency. With extra effort, the meaning of these sentences can usually be inferred, especially when the rest of the sentence is fluent (as in the first sentence in Table 6.1.

On the other hand, the feedback APEs try to balance the post-editor's request to include more information in the sentence against the goal of the decoder to produce fluent output. But the need for fluency also led to fewer modified sentences, particularly for phrase-level feedback. In cases where both APE approaches improve the adequacy, the feedback approach is better because it produces more fluent sentences. But in cases where the feedback approach does not modify the sentence, the rule-based approach can often still improve the adequacy of the translation at the expense of fluency.

6.4 APE for MT Systems with Fewer Deletions

One frequently asked question about APEs is, why not improve the MT system instead? Application developers typically do not build their own MT systems and have limited control over the systems they are using (for instance, Google or Bing Translate APIs). The main purpose of APEs it to adapt black-box (or glass box) MT systems to applications with specific needs. Even if one could build an application-specific MT system, no MT system is perfect: all open-domain MT systems produce translations with adequacy errors, which could potentially be addressed by APEs.

A closely related question is, what will happen to APEs as MT quality improves? Since the APEs adapt to the MT output, if the MT quality is very high, they should simply make fewer edits. Yet even as large-scale MT for major languages improves in quality, other situations still present challenges for MT. Low-resource language pairs, difficult new genres (Twitter, SMS, etc.) and devices with low computational power all present opportunities for APE.

In this follow-up experiment, we touch on both of these questions by repeating our earlier experiments on versions of the same MT systems that have been modified to drop fewer words. As described in section 6.3.4.2, the HiFST-noDrop system was produced by turning off the drop rules feature, so that it cannot arbitrarily delete words for which it already has translations, and re-tuning the system using MERT. The Moses-keep system is the same as the Moses system, except that it retains OOV words. The Moses-keep system is very similar to the Moses baseline system, while HiFST-noDrop is significantly different from HiFST, since it was completely re-tuned.

In both cases, the MT systems should produce translations with fewer outright deletions. This will not necessarily improve the MT quality: it will lead to improved adequacy if the MT system has correct translations for the previously dropped words, or degraded adequacy if the MT system mistranslates them. Evaluating performance on these MT systems will show how the APEs adapt to MT systems that make fewer deletion errors.

6.4.1 Results

For conciseness, we will refer to the HiFST-noDrop and Moses-keep MT systems as the **lessDrop** MT systems, and the HiFST and Moses MT systems as the **baseline** MT systems.

Number of Errors: Table 6.2 shows that the APEs detected much fewer errors in the lessDrop MT systems compared to the baseline MT systems. The percentage of sentences with detected errors dropped from 48% in HiFST to 27% in HiFST-noDrop for newswire; and from 69% to 38% for web. The decreases for Moses were less dramatic: 24% in Moses to 23% in Moses-keep for newswire, and 34% to 31% for web.

This meant that the APEs made fewer edits to the overall corpus: while the APEs modified 15% - 62% of the baseline MT sentences, they only modified 13% - 37% of the lessDrop MT sentences. The APEs successfully adapted to the MT output by editing less often.

Automatic Metrics: Figure 6.5 shows the automatic scores of all the MT systems. Results for the Moses and Moses-keep systems are very similar: Moses-keep has slightly lower Meteor-a scores, but otherwise the trends and scores are close. The feedback APE generally improves the baseline MT more than the rule-based APE, according to Meteor-a and TERp-a.

HiFST and HiFST-noDrop show more differences. The HiFST-noDrop system has a worse BLEU score, but better Meteor-a and TERp-a scores. While the feedback APE improves HiFST more than the rule-based APE, for HiFST-noDrop, we see the opposite: the rule-based APE has better Meteor-a and TERp-a scores.

Human Evaluation: The bottom half of Figure 6.6 shows results of the human-annotated contrastive adequacy evaluation. The major trends are the same as for the baseline MT systems: the APEs improve over the original MT 24% - 49% of the time, and they always improve adequacy more than they degrade it.

Both types of APEs have less of an impact on the lessDrop MT output than they do on the baseline MT output: across all APEs and genres, the percent of sentences with errors that were not edited or were judged “about the same” increased.

The rule-based APE improves adequacy less often on lessDrop MT output (38% - 49%) than on the baseline MT output (44% - 56%). It degrades adequacy more often on the HiFST-noDrop output (18% - 21%) than on the HiFST output (14% - 16%), though for Moses and Moses-keep, the numbers are similar.

The feedback APEs adapt to the lessDrop MT systems by frequently not editing. The corpus-based feedback APE does not edit 35% - 44% of HiFST-noDrop sentences with errors, as compared to only 10% - 16% for HiFST. Similarly, the phrase-based feedback chooses not to edit 31-43% of Moses-keep sentences with errors, compared to 26-39% for Moses. While this means that the feedback APEs improve adequacy less often for the lessDrop MT systems, they also degrade adequacy less often, so overall they still have a positive impact.

6.4.2 Discussion

To see how the APEs handled improved MT systems, we built lessDrop versions of each of the MT systems used in our previous experiments. HiFST-noDrop was significantly different than HiFST, in that it was not permitted to arbitrarily delete words that were not OOV. Moses-keep was very similar to Moses, except that it retained Buckwalter-transliterated OOV words in the output (which typically also affects how the rest of the sentence is translated).

The APE error detection algorithms identified fewer errors in the lessDrop MT output compared to the baseline MT output, which is what we hypothesized would happen, since the MT systems were making fewer deletions.

Across both lessDrop MT systems and both genres, the APEs still increased MT adequacy: the APEs were found more adequate than the original MT 24% - 49% of the time for HiFST-noDrop and 27% - 44% for Moses-keep. For all APEs, the percentage of time that post-editing improved adequacy was significantly higher than the percentage of time that adequacy was degraded.

Overall, the results show that when the MT systems are tuned to make fewer errors, the APEs successfully adapt by detecting fewer errors. Even though they post-edit fewer sentences, the APEs ultimately still have a positive impact on adequacy. This demonstrates that even as MT systems improve, APEs can still be useful for adapting MT output to the needs of specific applications.

6.5 Conclusions

In this chapter, we focused on improving intrinsic MT quality, specifically focusing on aspects of translation that were most crucial for CLQA. We introduced several techniques for automatically correcting the MT errors detected by our algorithms in the previous chapter. Whereas previous APEs focused primarily on translation fluency and grammaticality, our APEs targeted adequacy errors.

The relatively simple query-directed NE APE was able to use the CLQA query to correct NE mistranslations in CQLA translated results. The adequacy-oriented APEs addressed several additional types of adequacy errors, and drew on additional resources from the CLQA task to find translation corrections.

We described several techniques for adequacy-oriented APE: rule-based in addition to corpus-level and phrase-level feedback. Human evaluation showed that post-editing was effective in improving the adequacy of the original MT output 30-56% of the time, across two MT systems and two text genres. The APEs had a larger impact on the web text than the newswire, indicating that they are particularly useful for hard-to-translate genres.

Further evaluation of the APEs revealed a trade-off between fluency and control. The rule-based APE allowed control over which errors to correct and exactly how to correct them, but was limited to two basic edit operations that often led to disfluent sentences. The feedback APEs were more powerful than the rule-based APEs because they could edit the context around the correction as well as carry out multiple corrections at once. The feedback APEs produced sentences that were more fluent, but they relied on MT decoders that might or might not carry out the corrections. The corpus-level feedback APE was the least targeted, because suggestions passed to the MT system could affect any re-translated sentence, even those where the phrase was translated correctly. Surprisingly, it was still able to improve adequacy. The phrase-level feedback APE allowed more targeted error correction, yet had the least impact because it often ignored the corrections.

When the MT systems were modified to delete words less often, the APEs adapted well by detecting fewer errors and editing less often. The adequacy-oriented APEs still had a positive impact on translation adequacy, since improvements significantly outnumbered decreases in adequacy.

The APEs were motivated by the CLQA task, where adequacy errors can make correct answers appear incorrect after translation. Automatic post-editing is particularly suitable for task-embedded MT, where black box MT systems must be adapted to the needs of a specific task. In the next chapter, we describe a task-based evaluation of the adequacy-oriented APEs, which seeks to measure their impact on CLQA relevance.

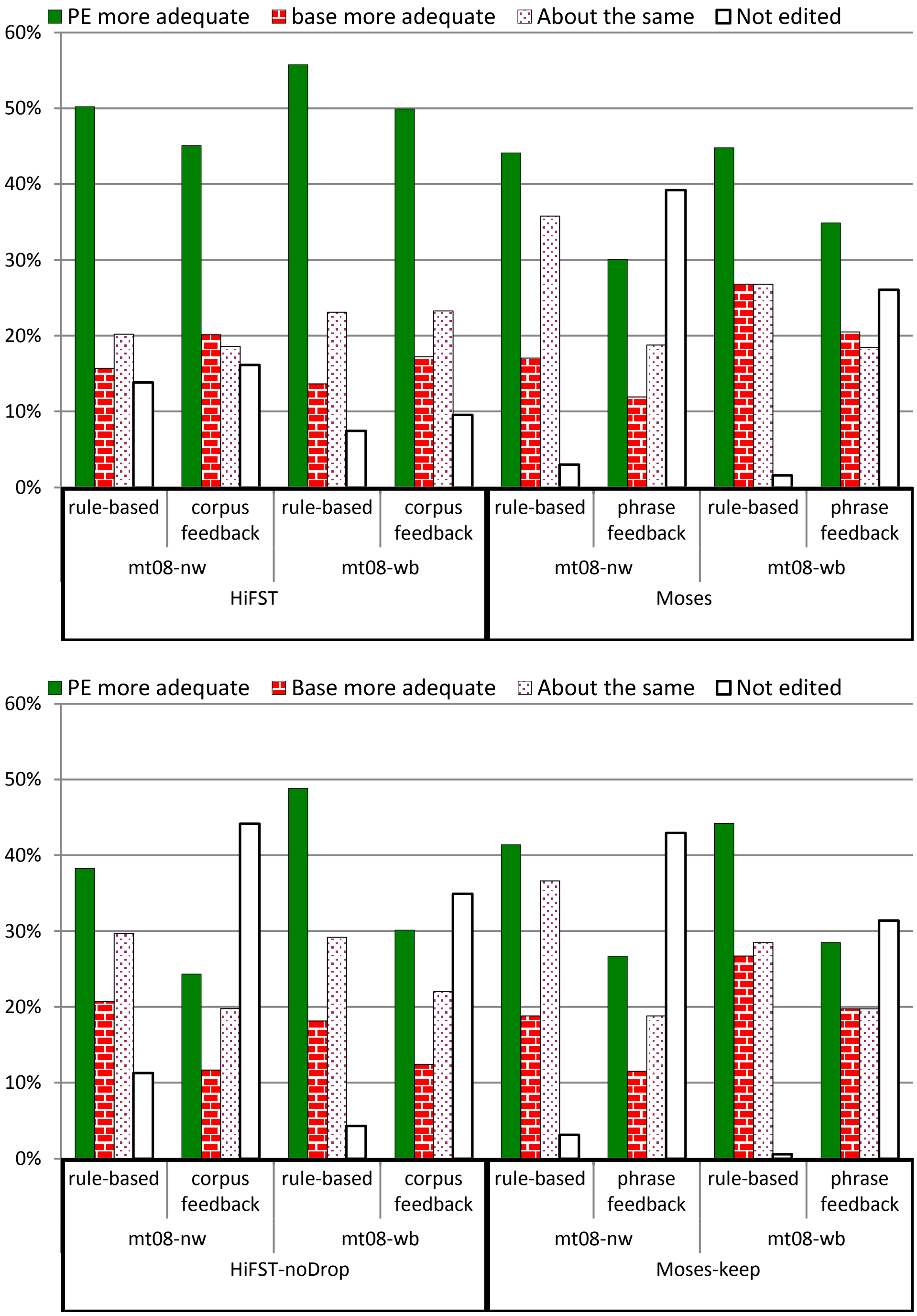


Figure 6.6: Percentage of post-edited sentences that were judged more adequate, less adequate or about the same as the original MT. "Not edited" is the percentage of sentences with errors that the APE decided not to modify.

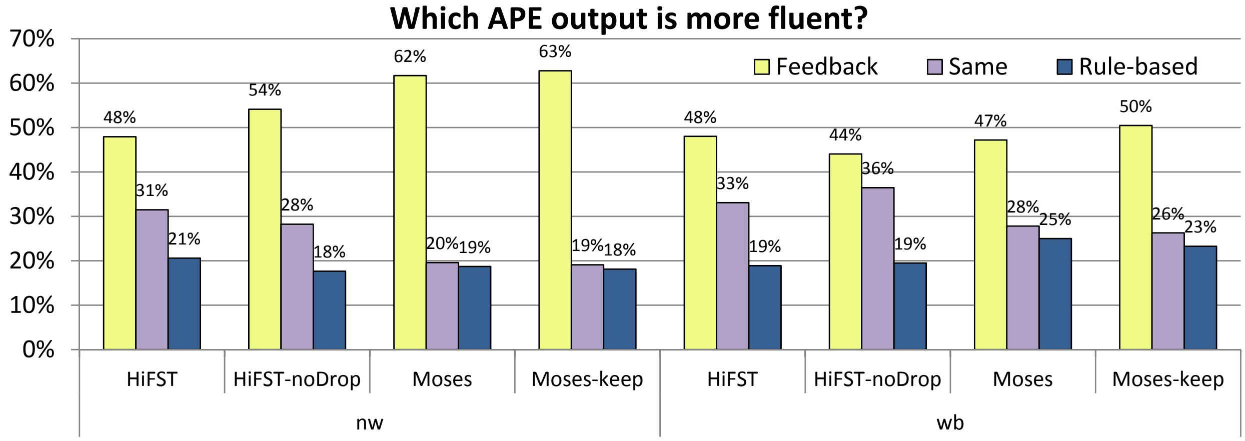


Figure 6.7: Results of the fluency evaluation, where annotators were asked to judge the relative fluency of the rule-based and feedback post-edited sentences. The feedback APE output is significantly more fluent across all genres and MT systems.

Chapter 7

Evaluating Task-Embedded MT

In the previous chapters, we have presented techniques motivated by and designed for use with the CLQA task, but we evaluated them intrinsically. The SMLIR model was evaluated on a CLQA task without result translation, and the APEs were evaluated in terms of MT metrics and human judgments of MT quality. In this chapter, we put them all together and evaluate SMLIR and the APEs in a task-based framework to see their ultimate impact on CLQA with result translation. The task that we evaluate can be thought of as a CLQA task with broad relevance guidelines, or a sentence-level CLIR task. In this chapter, we call it an IR task because we are interested in comparing different sentence-level retrieval models.

For conciseness, we refer to the CLIR task with result translation as *translingual* IR (translingual information retrieval (TLIR)), as discussed in Chapter 1. The TLIR task is defined as follows: given a query in language ℓ, and a corpus in language m, return relevant results from the corpus, translated from the document language m into the query language ℓ.[1] A system that carries out the TLIR task consists of a CLIR model, which is responsible for retrieving and ranking results, as well as an MT system, which is needed to translate results back to the query (user's) language.

To study the TLIR task, we created a TLIR evaluation corpus with relevance judgments on human translations as well as the output of the two MT systems from the last chapter (HiFST and Moses). The corpus is based on a standard Arabic-English MT test set. From an MT perspective, this corpus provides an extrinsic, task-based evaluation of MT output; from the viewpoint of CLIR,

[1] Using MT for CLIR query translation has been studied extensively [Gao *et al.*, 2001; Herbert *et al.*, 2011; Magdy and Jones, 2011] and is *not* the focus of this chapter.

it models a real-world application, where the end user can read the results of the CLIR system.

We use the TLIR corpus to compare various CLIR models and MT systems on a shared end-to-end task. We find some results that contradict intrinsic evaluations. Although HiFST and Moses have similar BLEU (bilingual evaluation understudy) scores, Moses performs better on the TLIR task. We compare two baseline CLIR models, and find that QT performs better than DT in an intrinsic retrieval evaluation, but the opposite is true in the TLIR evaluation, where both retrieval and result translation are taken into account. These results show that results from intrinsic evaluations do not always carry over to actual use of the system.

In addition to studying the complete TLIR task, we can also use the corpus to analyze two separate aspects of the task: retrieval accuracy and translated result understanding. By retrieval accuracy, we mean a traditional CLIR evaluation where result translation is not taken into account. Conversely, if we ignore retrieval accuracy, we can isolate how MT quality affects annotator's understanding of relevant results. MT errors can degrade each of these aspects of the TLIR task. When CLIR models retrieve over machine translated documents, MT errors can lead to recall errors, where the CLIR model fails to retrieve relevant results. On the other hand, even when a relevant result is found, if it is translated incorrectly it can appear irrelevant to the user. We refer to these two types of errors as *lost in retrieval* errors and *lost in translation* errors.

We use the TLIR corpus as a novel testbed to evaluate ways to address each of these types of errors. In Chapter 4, the SMLIR model outperformed baseline models in a CLIR evaluation without result translation; here we show that it also helps in a TLIR setting because it addresses many lost in retrieval errors as well as some lost in result translation errors. In Chapter 6, automatic post-editors (APEs) improved intrinsic MT quality; here we compare the impact of several adequacy-oriented APEs on lost in translation errors in the TLIR task.

In the next section, we describe the TLIR evaluation corpus and how we use it to analyze the effect of different MT systems and CLIR models on TLIR performance. After describing the experimental setup, we present an analysis of two baseline approaches to TLIR that suffer from lost in retrieval errors and lost in translation errors. We experiment with two retrieval models to address the lost in retrieval errors, and two APEs to address the lost in translation errors. By integrating CLIR and MT more closely and evaluating them in an end-to-end task, we are able to improve over the baseline pipelined approaches.

7.1 Related Work

CLIR allows users to find information in languages they do not know, but it has been described as "the problem of finding documents that you cannot read" [Oard *et al.*, 2008] because a separate MT system must be applied before the user can read the results. Shared CLIR tasks have been run by TREC, NTCIR and CLEF across a variety of language pairs and domains, and in most cases, relevance is judged in the document language. These evaluations are sufficient for evaluating retrieval models, but inadequate for assessing the usefulness of an end-to-end TLIR system: even if a system returns all the relevant results, if they are translated poorly, the user will not be able to understand them.

As with CLIR, MT systems are usually evaluated intrinsically with human judgments of adequacy and fluency, or with automatic metrics such as BLEU [Papineni *et al.*, 2002b], TERp [Snover *et al.*, 2009] and METEOR [Lavie and Agarwal, 2007]. These evaluations are aimed at open-domain, task-agnostic MT, so they seek to balance fluency and adequacy. In contrast, for *task-embedded MT*, where MT is used to translate the results of a cross-lingual application, adequacy may be more important than fluency (or vice versa).

Intrinsic measures of MT quality do not always correlate with the *usability* of MT output. As we described in Chapter 2, a very poor quality Russian-English MT system was recommended by 96% of the users simply because having any MT (even poor quality MT) was better than nothing, and having fast (but low-quality) translation was better than waiting for human translations, particularly when the MT helped filter out documents that did not need to be translated by humans [Church and Hovy, 1993]. As this example shows, it is important to consider the task context when evaluating the usability of MT systems.

This idea was championed by Church and Hovy [1993]: "If the application [for MT] is well-chosen, then it often becomes fairly clear how the system should be evaluated. Moreover, the evaluation is likely to make the system look good." The argument for task-based MT evaluation is that, since intrinsic evaluation is so difficult, task performance can be a proxy. Jones *et al.* [2007] created comprehension questions based on Arabic documents, and then gave the questions to humans along with the documents translated into English. Not surprisingly, humans did much better on reference translations than MT (95% vs. 74%), and questions with NEs were especially difficult due to mistranslations. HTER was not strongly correlated with comprehension, which suggests that the tests "are measuring a somewhat independent aspect of MT quality. . . [and] that not all MT errors are equally important," an observation that motivates the adequacy-oriented APEs.

Evaluating task-embedded MT is more difficult than evaluating either the task or the MT alone. In a CLIR evaluation, Hakkani-Tür *et al.* [2007] were unable to judge against MT output "because when the translation quality is poor that procedure tends to be too subjective and MT system-specific." In a large-scale translingual evaluation for the GALE distillation (question-answering) task, "inter-annotator consistency for relevance judgments on system-extracted (and machine translated) snippets range[d] from 59-89%" [Glenn *et al.*, 2011]. The Interactive Track at CLEF has also carried out TLIR evaluations [Oard and Gonzalo, 2002; Oard and Gonzalo, 2003], though the evaluations were relatively small in scale because they involved in-depth user studies.

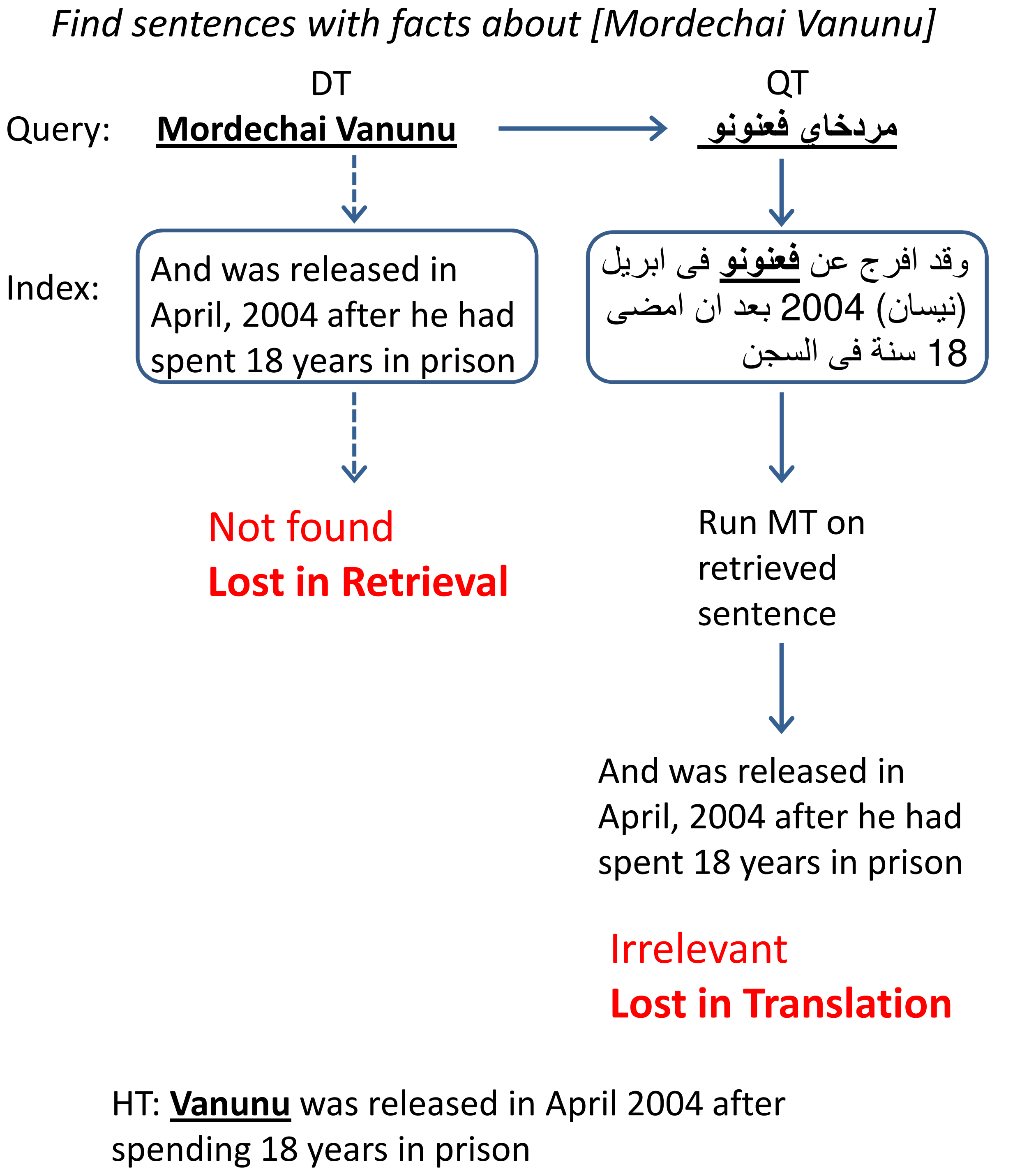
Despite the difficulties in evaluating TLIR, it is important because it offers opportunities for synergy between the CLIR and MT systems. Recent work by He and Wu [2011] showed that pseudo-relevance feedback on the translated results can improve query translation and translation probability estimation, which leads to improved retrieval accuracy. Like SMLIR, their model exploits the fact that results have to be translated before being shown to the user – something which is overlooked by CLIR models that focus only on source-language retrieval.

Just as CLIR evaluations focus on intrinsic measures of retrieval accuracy without taking into account result translation, MT evaluations are based on intrinsic measures of translation quality without considering the usability of MT output in applications such as CLIR. The fact that MT and CLIR are studied in isolation from each other leaves a large, under-studied gap surrounding the role of MT in real-life applications of CLIR. This chapter is an attempt to bridge that gap.

7.1.1 MT Errors and TLIR

MT errors affect TLIR differently depending on which model is used for retrieval. Two naive TLIR baseline systems are simple pipelines of independent MT and CLIR systems, which are demonstrated in Figure 7.1. In the document translation (DT) approach, the Arabic corpus is translated offline and then indexed in the query language, English. At query time, a monolingual search in the query language (English) is performed, and the MT sentences are retrieved.

In the query translation (QT) approach, the corpus is indexed in the document language (Arabic). At query time, the English query is translated into the document language (Arabic) and



Find sentences with facts about [Mordechai Vanunu]

DT

QT

Query: **Mordechai Vanunu**

مردخاي فعنونو

Index:

And was released in April, 2004 after he had spent 18 years in prison

وقد افرج عن **فعنونو** فى ابريل
(نيسان) 2004 بعد ان امضى
18 سنة فى السجن

Not found
Lost in Retrieval

Run MT on retrieved sentence

And was released in April, 2004 after he had spent 18 years in prison

Irrelevant
Lost in Translation

HT: **Vanunu** was released in April 2004 after spending 18 years in prison

Figure 7.1: The results of running the query "Provide information about Mordechai Vanunu" against two different TLIR pipelines. The NE deletion in MT affects the QT and DT retrieval models differently.

Arabic sentences are retrieved. Then MT is run to translate the results into the query language (English). (The QT and DT models are discussed in more detail in Chapter 4.)

Figure 7.1 shows how an MT error can affect each of these retrieval models in different ways. In the DT approach, the indexed MT sentence is missing the NE, so at query time, it cannot be retrieved by the English query. In terms of CLIR, this is a recall error. We refer to this as a *lost in retrieval* error.

In the QT approach, the English query is translated into Arabic using MT and additional resources. In the QT pipeline in Figure 7.1, the Arabic sentence is retrieved and then translated using MT. However, when the user sees the sentence, it appears irrelevant because there is no mention of the query. In CLIR, this would not be considered an error, since the sentence is relevant in Arabic. In TLIR, it is a precision error because an irrelevant result was returned. But the error is due only to a loss in MT adequacy and not due to the retrieval model. We refer to this as a *lost in translation* error.

When MT errors occur, they cause lost in retrieval errors in the DT model and any retrieval model that relies on translated documents. The QT model avoids lost in retrieval errors, but is still affected by lost in translation errors, where the relevant information is garbled or missing due to MT. In our experiments, we quantify these errors and their impact on the end-to-end TLIR system, and evaluate two approaches to mitigating these errors.

7.2 TLIR Evaluation Corpus

In the English-Arabic TLIR task, the system is presented with an English query and a set of Arabic documents. The goal is to find all Arabic sentences relevant to the query and return them machine translated into English. Each experimental setting consists of a CLIR model and an MT system; we experiment with four CLIR models and three types of translations (human translations, HiFST and Moses). As in a standard IR evaluation, we run a set of English test queries on all systems and pool the top k returned translated sentences. Then we ask annotators to judge each query-sentence pair as Relevant or Not Relevant. We aggregate the judgments using mean average precision (MAP), defined below.

One of the challenges in evaluating TLIR is finding an appropriate evaluation corpus. The ideal corpus would enable us to measure both retrieval relevance and MT quality, so we would like a corpus that has human translations (HTs) as well as query-result relevance judgments. CLIR corpora are very large datasets with query-document pairs annotated in the document language. It is expensive to translate such large corpora with MT and infeasible to translate manually. In contrast, MT test sets have reference translations, but are small in comparison.

To create the TLIR evaluation corpus, we augment a standard MT test set (NIST MT08 Arabic-English) with queries and sentence-level relevance judgments: we create 94 queries for 813 sentences in the newswire (NW) corpus, and 77 queries for 547 sentences in the web (WB) corpus. This is a tiny corpus by CLIR standards. We are explicitly trading off the size of the corpus in order to have a corpus with reference translations because we are ultimately interested in the impact of MT errors on TLIR. For each query-sentence pair, we collect two types of relevance judgments:

HT relevance: Relevance judgments on the HT of each sentence. Human translation is the upper bound for TLIR result translation quality.

MT relevance: Relevance judgments on each MT version of each sentence. Annotators judge MT relevance without seeing the HT, so sentences that are garbled during MT are judged irrelevant even when the HT version of the sentence is actually relevant.

In the rest of this section, we define the task for the TLIR evaluation corpus. First we describe the types of queries we are interested in. Then we describe the metric we use to evaluate TLIR relevance. Finally, we describe how we use the HT and MT relevance judgments to isolate lost in translation errors and lost in retrieval errors.

7.2.1 TLIR Task

Our task was inspired by the GALE distillation task, which was an open-ended, template-based CLQA task with result translation, and was motivated by the needs of intelligence analysts. Since the queries in that task focused on NEs, we chose to base our queries on NEs. However, we are using untrained annotators, so we chose to focus on a broader type of query, similar to the "Provide information on NE" template.

Specifically, annotators were asked "Which sentences contain facts about NE?" The instructions emphasized that merely mentioning the NE was neither necessary nor sufficient for relevance. For instance, the sentence "The US President visited Iran in 2012." is relevant to a query about Barack

Obama, even though his name does not explicitly appear in the sentence. The same sentence is not relevant to a query about the US, even though the US is mentioned, because there is no fact about the US.

This type of query is interesting because it requires sentence-level MT understanding. If a sentence is completely garbled or missing a verb, it does not have a fact, so it is not relevant. Even if the NE in the query is mistranslated, the annotator may be able to use context clues in the sentence to determine relevance.

7.2.2 Evaluation Metrics

For results using HT relevance, we report the mean average precision (MAP), which takes into account both recall and precision [Manning *et al.*, 2008]. MAP is a standard metric for evaluating ranked results that is commonly used in IR evaluations. (We review the MAP formula from Chapter 2 here because we modify it further below.) This metric summarizes the overall performance of the system by taking the mean over all queries of the average precision across all levels of recall. More formally, for a query q, denote by Rel_q^{HT} the set of all HT sentences that are relevant to q. Then for each result d_q in a set of n ranked results, relevance, precision at k and average precision are defined as:

$$rel^{HT}(d_q) = \begin{cases} 1, & \text{if } d_q \in Rel_q^{HT} \\ 0, & \text{otherwise} \end{cases}$$

$$Prec(k) = \frac{\sum_{i=1}^{k} rel^{HT}(d_q^i)}{k}$$

$$AvePrec = \sum_{k=1}^{n} \frac{(Prec(k) \times rel^{HT}(d_q^k))}{|Rel_q^{HT}|}$$

MAP is the average of $AvePrec$ over all the queries.

When evaluating TLIR, a result is relevant only if it is actually relevant (in HT) *and* is perceived as relevant by the user (in MT). Let Rel_q^{MT} represent the set of all MT sentences that are relevant to q. We will consider a sentence d_q is relevant to a query q only if it is in Rel_q^{MT} in addition to

Rel_q^{HT}. Formally, the relevance, precision and average precision are defined as:

$$rel^{MT}(d_q) = \begin{cases} 1, & \text{if } d_q \in (Rel_q^{MT} \cap Rel_q^{HT}) \\ 0, & \text{otherwise} \end{cases}$$

$$Prec(k) = \frac{\sum_{i=1}^{k} rel^{MT}(d_q^i)}{k}$$

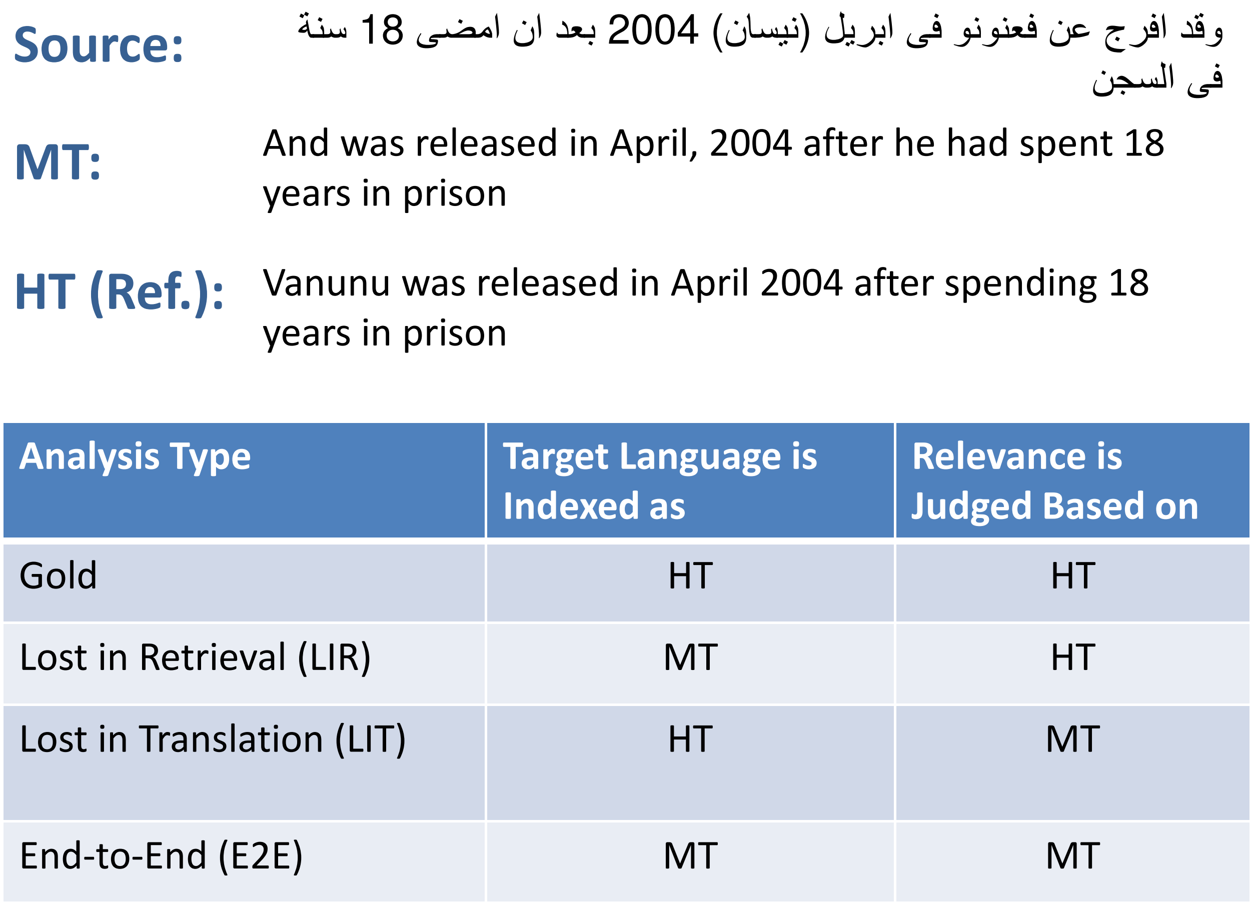
$$AvePrec = \sum_{k=1}^{n} \frac{(Prec(k) \times rel^{MT}(d_q^k))}{|Rel_q^{HT}|}$$

For a given retrieval model and MT test set, HT MAP measures how good the retrieval model is, independent of whether the end-user can understand the results, while MAP using MT relevance measures both retrieval and result understanding. In other words, MT MAP is essentially regular (HT) MAP with a penalty for results that cannot be understood in translation.

7.2.3 Analysis Methods

Since we have relevance judgments on both HT and MT, we can compare the TLIR performance upper bound (on HT) to the TLIR performance using MT, and quantify the percent lost due to MT. We can also use these two types of relevance judgments to isolate the effects of MT on retrieval and on result translation separately. In each TLIR setting, the MT is used in two ways: during retrieval, most of the CLIR models use the translated corpus to retrieve and/or rank the results; during relevance annotation, the translated results are presented to the user. By varying retrieval and relevance annotation between HT and MT, we can analyze our results for each setting four different ways:

- **Gold (HT)**: HT is used for both retrieval and relevance annotation. This setting is an upper bound on TLIR system performance.
- **Lost in Retrieval**: MT is used for retrieval only; the results are annotated in HT, to remove the influence of MT errors on result understanding. This setting is similar to standard CLIR evaluations where result translation is not taken into account.
- **Lost in Translation**: HT is used for retrieval, but results are judged in MT. This setting ignores the impact of MT errors on retrieval accuracy, and focuses on sentences that should be relevant but are judged irrelevant due to errors in result translation.



Source: وقد افرج عن فعنونو فى ابريل (نيسان) 2004 بعد ان امضى 18 سنة
فى السجن

MT: And was released in April, 2004 after he had spent 18 years in prison

HT (Ref.): Vanunu was released in April 2004 after spending 18 years in prison

Analysis Type	Target Language is Indexed as	Relevance is Judged Based on
Gold	HT	HT
Lost in Retrieval (LIR)	MT	HT
Lost in Translation (LIT)	HT	MT
End-to-End (E2E)	MT	MT

Figure 7.2: The four different types of analysis we perform on each TLIR system, using different combinations of human translation (HT) and MT. Indexing using HT gives an upper bound for retrieval, while judging relevance using HT gives an upper bound for MT. (Note that some retrieval models do not index in the target language; Table 7.1 lists the indexed language for each model.)

- **End-to-end (MT)**: MT is used for both retrieval and relevance annotation. This setting represents the full end-to-end TLIR system, where MT has an impact on both retrieval and result understanding.

Retrieval Model	**Indexed Language**
Query translation (QT)	Source
Document translation (DT)	Target (MT or HT)
SMLIR	Source and Target (MT or HT)
QT-rerank	Source

Table 7.1: In our experiments, Arabic is the source language and English is the target language. QT and DT are the baseline models, and SMLIR and QT-rerank are hybrid models. In DT, SMLIR and QT-rerank, "target" may refer to either HT or MT, depending on the analysis type (see Figure 7.2). Note that even though QT-rerank does not index English, it does use English to re-rank the retrieved results.

7.3 Experiments

7.3.1 Query Extraction

Queries were created by running the Stanford NE recognizer [Finkel *et al.*, 2005] on one of the human translations. This list of all possible NE queries for the corpus was filtered to remove near-duplicates and incorrectly tagged phrases. After relevance annotation, the queries were further filtered to remove queries that had one or zero results. The average number of relevant sentences per query was 8 for NW and 5 for WB. The MT08 corpus statistics are shown in Table 7.2.

7.3.2 MT systems

We use two pre-existing state-of-the art MT systems, HiFST and Moses. These are the same systems we used in Chapter 6 to evaluate the APEs, and they are also described in detail in Chapter 3. HiFST is a hierarchical, phrase-based MT system trained on 5.9 M parallel sentences (150M words per language) with a 5-gram language model trained on 6.6 B words of English text; Moses is a non-hierarchical, phrase-based MT system trained on 3.2M parallel sentences (60M words per language), and uses a 5-gram LM trained on Gigaword 4 (2.9 B words). On the NW corpus, the BLEU scores for HiFST and Moses are 51.32 and 51.23, respectively; and on WB, 36.15 and 37.60, respectively. Based on these scores, HiFST and Moses are comparable in quality to each other and competitive with systems in the MT08 competition.

7.3.3 Relevance Annotation

The same HT that was used to create the queries was also used as the gold standard for collecting relevance judgments. While an ideal CLIR evaluation would have gold relevance measured in the source language, Arabic annotations were difficult to get, and we opted for a large quantity of reference annotations instead of a small set of source language annotations. The relevance judgments were done on Amazon Mechanical Turk (AMT) using Crowdflower[2] to filter for trusted workers. Relevance judgments on AMT have been shown to have high agreement with TREC raters [Alonso and Baeza-Yates, 2011]. We pooled the top-10 results of each TLIR run and crowd-sourced 3 relevance judgments each on three versions of each query-sentence pair: the HT and both MT versions. The annotation instructions are shown in Appendix Figure B.10, and two examples of annotating MT are shown in Appendix Figures B.11 and B.12.

7.3.4 Baseline Systems

Document Translation For DT, the translated English sentences are indexed, and the query is run against the indexed translations. An index is built for each version of the corpus: HT, HiFST and Moses.

Query Translation For QT, the Arabic source sentences are indexed, and the English query is translated into an Arabic query in order to do monolingual search in Arabic. The QT method uses a cascaded approach to query translation, and then converts the translated query into a structured query [Pirkola, 1998]. The first step in translation is a NE dictionary built from Wikipedia [Ferrández *et al.*, 2007], the CIA world factbook and the NEs from the Buckwalter analyzer dictionary [Buckwalter, 2004]. Since all of the queries are NEs, this dictionary is a high-precision, but low recall resource. If the translation is not found, a phrase table from Moses is used as a dictionary. The final back-off searches a large corpus of machine translated documents, which is a lower precision resource. If the translation is still not found, the original query is expanded using synonyms extracted from Wikipedia, and then the cascaded translation is applied again.

[2] http://www.crowdflower.com

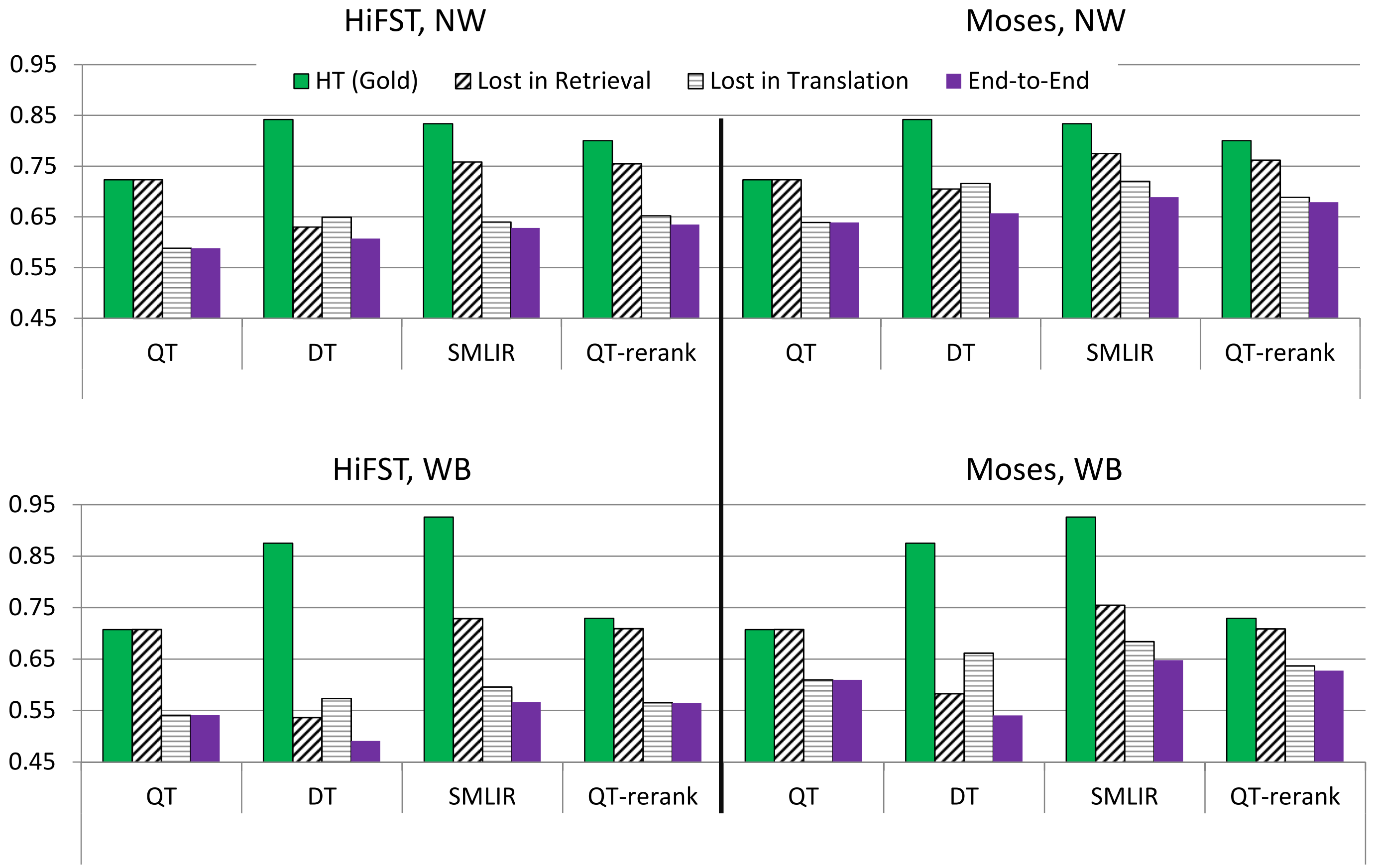


Figure 7.3: The MAP of different combinations of MT systems and retrieval models using the analysis methods from Figure 7.2. The HT (gold) setting is an upper bound using only reference translations for retrieval and relevance annotation. The lost in retrieval setting ignores the impact of MT on result translation, while the lost in translation setting ignores the impact of MT on retrieval. The end-to-end setting uses only MT for both retrieval and relevance annotation.

7.4 Analysis of Baselines

Figure 7.3 shows results from all experimental settings; in this section we discuss the results for QT and DT. All of the models are evaluated on the newswire and web genres (the top and bottom charts, respectively) and HiFST and Moses (left and right, respectively). Each setting is analyzed four different ways, summarized in Figure 7.2.

This analysis demonstrates the strengths and weaknesses of the different models. The QT model is unaffected by retrieval errors, since retrieval is done in the document language only (MAP is the same for Gold and Lost in Retrieval bars), but is significantly degraded by result translation errors. For instance, for QT in the HiFST NW setting, the MAP is 19% lower when results are judged in MT instead of HT (Lost in Translation vs. Gold).

On the other hand, lost in retrieval errors have a significant impact on the DT model because it fails to retrieve sentences with MT errors. Across various conditions, MAP for DT decreases by 16-39% when retrieval is done using MT instead of HT (Lost in Retrieval vs. Gold).

In all cases, QT has higher MAP than DT in the Lost in Retrieval setting, and the reverse is true in the Lost in Translation setting. If we ignore result translations, as in a standard CLIR evaluation, the QT model is better than DT. When we ignore MT errors during retrieval but annotate relevance in MT, the DT model is better than QT. In other words, translated sentences that “look” relevant are ranked higher by DT than by QT.

In the End-to-end evaluation, where MT is used for both retrieval and relevance annotation, the results depend on the genre. In the NW genre, DT has higher end-to-end MAP than QT. In the WB genre, even though DT has higher MAP than QT when HT is used for retrieval (the lost in translation setting), its much lower retrieval accuracy (the lost in retrieval setting) ultimately makes the end-to-end DT MAP lower than QT. We attribute the poor performance of DT on WB to the difficulty of the genre: the text is informal and harder to translate, and as a result, there are more MT errors. If we consider the drop from Gold to End-to-end, we can see that it is larger in the WB genre than in NW (across all conditions). The relatively worse MT quality has a measurable impact on the TLIR task.

Across all settings, Moses has higher MAP than HiFST. Although both systems have similar BLEU scores on the evaluation corpora, when we looked at the examples, we found that Moses had much better NE translation, which is crucial for this task. The extrinsic TLIR evaluation shows

Corpus	Sents	Sent. Len. (Ar/HT)	Queries	Rel. Sents per query
MT08 NW	813	26.8 / 31.5	94	8.1
MT08 WB	547	29.0 / 36.2	77	5.1

Table 7.2: Statistics of the TLIR evaluation corpora: the number of sentences, the average number of words per sentence in Arabic and the HT, the number of queries, and the average number of relevant sentences per query.

the relative usefulness of the MT systems, which the intrinsic measures of MT quality could not capture.

Similarly, results of the intrinsic CLIR evaluation do not always match results from the full end-to-end TLIR task. For instance, in the NW genre, QT has higher MAP than DT in the CLIR evaluation (Lost in Retrieval), but DT has higher (translated) MAP than QT in the TLIR evaluation (End-to-End). This highlights the limitations of evaluating CLIR models without taking result translation into account.

7.5 Lost and Found in Retrieval

The baseline QT and DT models are both severely affected by errors in MT. A better retrieval model would retrieve all the relevant results, but rank the translations that appear relevant the highest. We experiment with the SMLIR model from Chapter 4 and a closely related model, QT-rerank.

SMLIR: Recall that in the SMLIR model, each sentence is indexed as a bilingual sentence with different fields for each language, and the structured query is composed of both query-language and document-language terms. In a CLIR evaluation without result translation, SMLIR outperformed both QT and DT. (Note that in Chapter 4, we evaluated the model in the document language (Chinese), whereas here we are concerned with relevance judgments on translated results.)

QT-rerank: One of the major disadvantages of the SMLIR model is that it requires a full corpus translation. As we have seen, the DT model is useful for finding sentences that are relevant in translation, but it has lower recall than QT. Actually, the results from DT are usually a subset

	MT	QT	DT	SMLIR
Correct MT	Vanunu	1	1	1
OOV	fEnwnw	1	-	2
Deletion	(empty)	1	-	2

Table 7.3: Given a sentence with a single Arabic NE, the rows show three different MT outputs and their ranking according to the different CLIR models. DT retrieves only the correct MT. QT retrieves all of them, but cannot distinguish between them. SMLIR retrieves all of them and ranks the correct MT highest (as would QT-rerank).

of the results from QT. The QT-rerank model retrieves results using QT only, then translates the results using an online MT system, and finally re-ranks the translated results using DT. This approach does not require corpus translation, though more work must be done at query time to translate the results and then re-rank.

Table 7.1 summarizes the different CLIR models, and Table 7.3 compares the effect of an MT error on result ranking.

7.5.1 Results

The hybrid models use the complementary strengths of QT and DT to their advantage to mitigate lost in retrieval errors. In the lost in retrieval setting, the SMLIR model always does better than either QT or DT alone. The document language part of the query ensures that relevant sentences are not missed during retrieval due to MT errors, and the query language part of the query gives higher rank to retrieved sentences that are translated correctly.

Although the hybrid models were designed to improve retrieval only, SMLIR also helps lost in translation errors in the web genre. This is because sentences that match in both the query language and the document language are ranked higher than sentences that match in only one language, so the highest ranking sentences tend to be those that are actually relevant in translation.

By reducing lost in retrieval errors and some lost in translation errors, the SMLIR model is able to achieve higher end-to-end MAP than either QT or DT alone across all settings.

The QT-rerank model also has better end-to-end performance then either DT or QT alone.

MT	And was released in April, 2004 after he had spent 18 years in prison
Rule-based APE	And was vanunu released in April, 2004 after he had spent 18 years in prison
Feedback APE	Vanunu was released in April, 2004 after he had spent 18 years in prison

Table 7.4: The result of automatically post-editing the MT from Figure 7.1 (repeated from the previous chapter for clarity). While both APEs insert the correct NE, the feedback APE produces a more fluent sentence.

QT-rerank does not use the query language during the first-pass retrieval, only during the second pass reranking. This means that sentences ranked low by QT but high by DT may be missed by QT-rerank, but will be returned by SMLIR. For HiFST, QT-rerank does as well or better than SMLIR, while for Moses, SMLIR has better end-to-end performance. In other words, QT-rerank has the advantages of SMLIR without the overhead of translating the full corpus.

7.6 Lost and Found in Result Translation

Figure 7.4 shows the Lost in Translation errors (where sentences were judged relevant in HT but not in MT) as a percent of query-sentence pairs. As in Figure 7.3, the WB genre has a higher error rate than the NW and Moses makes fewer errors than HiFST. In these cases, even a retrieval model with perfect recall would not help, since the problem is in the MT only.

We experiment with using the adequacy-oriented APEs from Chapter 6 to correct MT errors that affect TLIR relevance. Experiments in the previous chapter showed that 30-56% of the translations edited by the APEs had improved adequacy, and adequacy was degraded less than half as often. In this chapter, we use the TLIR evaluation corpus to see whether the improvement in adequacy leads to an improvement in TLIR relevance.

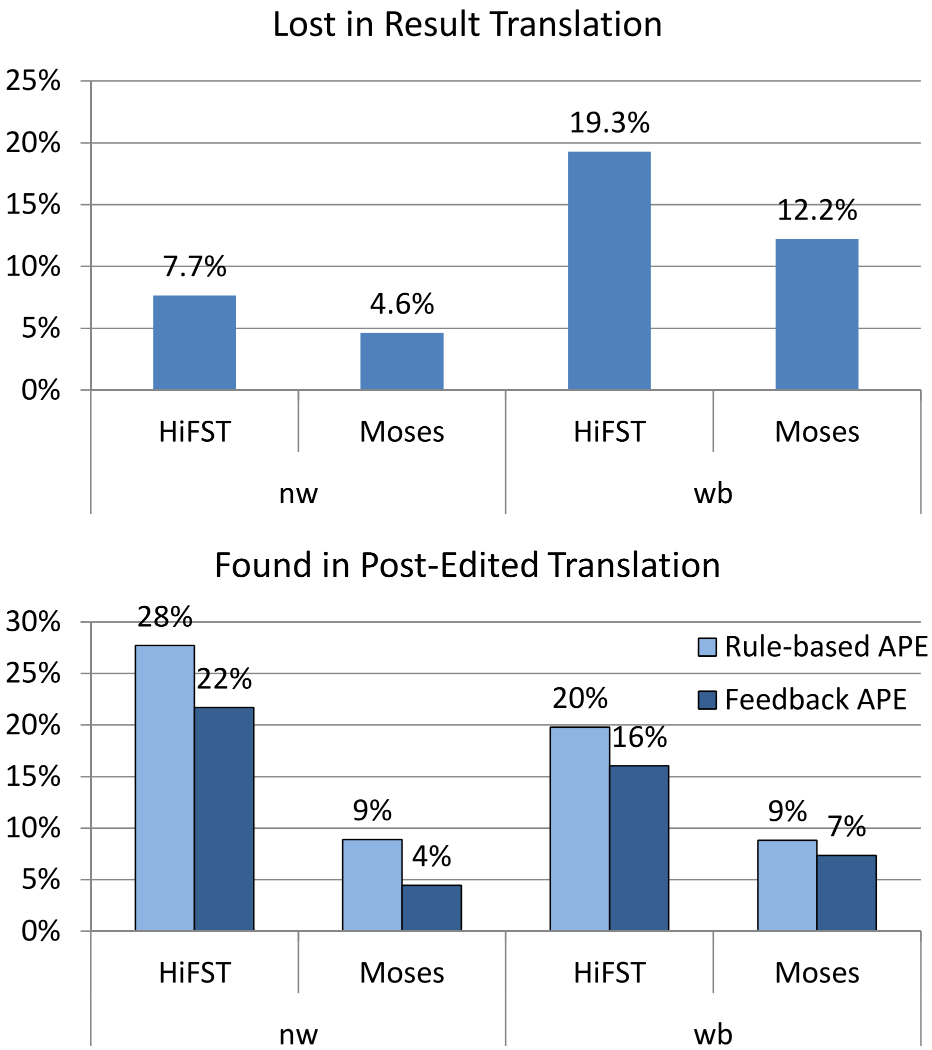


Figure 7.4: a) The percent of sentence-query pairs with lost in translation errors, where a relevant HT result became irrelevant in MT. b) The percent of lost in translation errors that were corrected by each APE.

7.6.1 Results

The rule-based and feedback APEs were run on all MT sentences. We collected 5 relevance judgments on post-edited sentences that were lost in translation errors (relevant in HT but not in MT), as well as on a sample of post-edited sentences that were relevant in both MT and HT, to measure the effect of errors made by the APEs.

Figure 7.4 shows the percent of errors that were corrected by the APEs – sentence-query pairs that were relevant in HT, irrelevant in MT, and relevant in post-edited MT. The APEs had a positive impact on result relevance, with 16-28% of HiFST errors and 4-9% of Moses errors corrected. Since the rule-based APE is more aggressive than the feedback APE in editing more often, it corrects more errors. Both APEs rarely caused a relevant MT sentence to look irrelevant (less than 0.01% of the time). These results show that adequacy improvements in the translated results can lead to improved end-to-end TLIR relevance.

7.7 Conclusions

We presented a TLIR evaluation that quantified the impact of result translation on retrieval and translated relevance, as well as on the end-to-end system. The QT and DT baselines that simply pipelined CLIR and MT were significantly degraded by MT errors compared to HT upper bounds. The TLIR evaluation corpus that we created was a crucial resource for this analysis, and also provided an experimental framework for testing proposed improvements over the baseline CLIR and MT systems.

The standard QT CLIR model had higher recall than DT, but DT had better relevance in translation. Our hybrid model SMLIR (introduced in Chapter 4) exploited these complementary advantages and did better than either alone. One limitation of SMLIR is that it requires the entire corpus to be machine translated prior to indexing. We presented a less resource-intensive hybrid model, QT-rerank, that reranks QT results by the DT model. QT-rerank performed as well as SMLIR, but is able to use online result translation instead of full corpus translation.

We also experimented with modifying the MT output to improve result relevance. We applied the two adequacy-oriented APEs introduced in Chapter 7 to sentences that were relevant in HT, but irrelevant in MT. They corrected adequacy errors such as missing content words and mistranslated NEs, which were especially important for our TLIR task, since all of the queries were NEs. The APEs improved sentence relevance 4-28% of the time and rarely degraded relevance.

We had previously evaluated the SMLIR model intrinsically by evaluating relevance without result translation, and we similarly evaluated the APEs in terms of MT quality. Evaluating TLIR results gives an extrinsic measure of both CLIR and MT system usefulness that goes beyond intrinsic quality. While the QT model performed better than DT on the CLIR task (without result translation), the DT model was better than QT on the end-to-end TLIR task, when results were judged in translation. Task-based evaluation of MT highlights specific errors that affect translation usefulness that are not evident from standard MT evaluations. For example, HiFST had a slightly higher BLEU score than Moses in the newswire genre, but HiFST had worse translated relevance than Moses because Moses was better at translating NEs.

The SMLIR model and the APE techniques were both designed with the TLIR task in mind. Unlike most CLIR models, SMLIR assumes that the results need to be translated before being displayed to the user, and exploits result translation to improve retrieval. The APEs take advantage of resources available from the task context to flag MT errors and to find better translations for those errors. By taking a holistic view of the TLIR task, we were able to improve both retrieval and MT. The task-based evaluation showed that these intrinsic improvements resulted in a better end-to-end TLIR system.

Chapter 8

Conclusions

While English was once the dominant language on the Internet, increasing global Internet penetration has led to increasing linguistic diversity on the web. Yet there is still a large gap between the languages spoken by Internet users and the languages that content is written in. Both of these trends underscore the essential role of MT in accessing information on the Internet.

At the same time, the pace of digital content creation keeps accelerating, so simply accessing information is not sufficient. Users need to be able to find, explore, analyze and use information – even when the information is in another language. Cross-lingual information retrieval (CLIR) and cross-lingual question answering (CLQA) help users find information in other languages about specific facts and topics.

Modern MT systems can produce high-quality translations for a variety of language pairs and genres, and modern CLIR systems can successfully retrieve foreign language results with high accuracy. However, as we have seen in this thesis, simply connecting a CLIR system to an MT system does not yield optimal end-to-end results. Relevant results retrieved by the CLIR system may appear irrelevant to the end user due to MT errors. Conversely, an MT system optimized for intrinsic measures of general MT quality may not be useful for tasks where specific aspects of translation are more important than others – in particular, in CLQA, translation adequacy is crucial. The fact that MT and CLIR are studied in isolation from each other leaves open many questions about the role of MT in real-life applications of CLIR – questions which this thesis seeks to answer.

In the next section, we provide an overview of the thesis contributions and describe how each contribution (in bold) relates to the overall problem. Following the overview, we delve into each contribution in detail and describe the impact of each. Then we present limitations, and finally we conclude with future directions in task-embedded MT.

8.1 Overview of Contributions

In this thesis, we have looked at the issues involved in using pre-existing MT systems for searching across and through different languages, since we ran experiments in both cross-lingual QA ("across" the language barrier once, with no result translation) and translingual QA (from the query language "through" the document language, and back into the query language). Our **task-oriented MT error analyses** showed that task-agnostic MT systems often do not produce suitable translations for the CLQA task. Open domain MT systems typically seek to balance fluency and adequacy, but for the CLQA task, adequacy is more important. The MT errors that most degraded translation adequacy for the CLQA task were NE mistranslations and missing or deleted content words.

MT errors led to decreased CLIR recall when models that retrieved based only on MT could not return relevant results due to missing or mistranslated NEs. Our **novel bilingual models for CLIR**, SMLIR and QT-rerank, address these errors by using both source- and target-language information to retrieve and rank relevant results. By integrating MT more closely into the CLIR model, we were able to improve retrieval accuracy.

Even when relevant results were found during CLIR, if the MT was garbled, the results could still be perceived irrelevant by the end user. We presented **methods for detecting and correcting MT errors** via automatic post-editing (APE). The rule-based and feedback APEs addressed adequacy errors in MT output, leading to improved adequacy over the baseline MT across two different MT systems and two evaluation corpora. The APEs focused on adequacy errors that our error analyses found to be most detrimental to CLQA performance, namely, NE mistranslation and missing and deleted content words. They used the task context to identify errors and find improved translations. By applying task-aware APEs to pre-existing task-agnostic MT systems, we were able to adapt the MT output to the needs of the CLQA task.

We showed that integrating MT and CLIR more closely could improve intrinsic measures of CLIR and MT quality, but evaluating the models extrinsically presented additional challenges since there was no existing test data for the "translingual" QA or IR tasks. We created a **translingual information retrieval (TLIR) evaluation corpus** over an existing MT evaluation corpus and collected human judgments of relevance over query-sentence pairs, annotating the human translation and two different machine translations of each sentence. Based on this corpus, we came up with an analysis framework for isolating the impact of MT errors on CLIR and on result understanding, as well as evaluating the end-to-end TLIR task.

Finally, we used the TLIR corpus to carry out a **task-embedded MT evaluation**, where we evaluated our CLIR and APE models extrinsically. We found that the improvements made by the SMLIR and QT-rerank models in the CLIR task carried over to the TLIR task. The extrinsic evaluation demonstrated more benefits of these bilingual models: in addition to improving retrieval accuracy, they often had a positive impact on perceived relevance because results that were translated well were ranked higher. We also found that the APEs successfully addressed many of the "lost in translation" errors, where results that were relevant in human translation were perceived irrelevant in MT. The extrinsic evaluation was an important complement to the intrinsic evaluations, as some of the results from the intrinsic evaluations did not hold in the end-to-end TLIR evaluation.

8.2 Contributions

1. Task-Oriented MT Error Analyses: In Chapter 3, we presented MT error analyses for three different SMT systems that represent the state-of-the-art spanning five years: the systems were built in 2006, 2009 and 2011. In the error analyses, we described the types of MT errors that most impacted the CLQA task, investigated how they arose, quantified how prevalent they were and documented their impact on translation adequacy. Our findings motivated our research in the rest of the thesis:

- Adequacy errors are prevalent in MT output.
- Named entity (NE)s are crucial for CLQA/CLIR, but very challenging for MT systems to translate correctly.
- Missing content words are common in MT and significantly degrade translation adequacy.

- Adequacy errors in MT can significantly decrease CLIR and CLQA recall.

2. Novel Bilingual Models for CLIR: In Chapter 4, we introduced two new CLIR models specifically designed for CLIR with result translation: the Simultaneous Multilingual Information Retrieval (SMLIR) model and the query translation re-rank (QT-rerank) model. These models have the following advantages for the TLIR task:

- **Bidirectional Translation**: By doing translation in both directions (document-to-query language and query-to-document language), these hybrid models exploit the complementary advantages of query translation (QT) and document translation (DT). The QT approach is more flexible, since multiple translations of each query term can be incorporated into a structured query, but the DT approach may produce better translations due to the larger (sentence-level) context.

- **Bilingual Retrieval**: By doing retrieval in both languages, the SMLIR model can find results that match in either language while promoting results that match in both languages. This addresses the recall problems faced by the DT approach that we identified in our error analysis. If a query term is mistranslated by the MT system during document translation, it can still match the sentence in the source (document) language via query translation because QT uses dictionary-based translation in addition to SMT dictionaries. Results that match via both QT and DT are more likely to be relevant, and are accordingly ranked higher than results that only match in one language. Unlike prior hybrid models that combine the results of QT and DT via tuned parameters, the SMLIR approach integrates both approaches into a single coherent model, where both the queries and documents are represented bilingually.

- **Ranking with Translated Results**: By including result translations during ranking, both models rank results with correctly translated query terms higher than results with MT errors. In other words, results with MT errors that are likely to be perceived irrelevant in translation will be ranked lower than results without MT errors. For instance, if two sentences mention the query term Obama in Arabic, but an MT error in one sentence causes the name to be deleted in the English result translation, then the sentence translation containing Obama will be ranked higher. This problem is not commonly discussed in CLIR research, since CLIR models are evaluated without result translation, yet it has a significant impact on real-life applications of CLIR.

- **TLIR Using Online or Offline MT**: By using DT to do result ranking (but not retrieval), the QT-rerank approach gets most of the advantages of the SMLIR approach without the cost of having to translate the entire corpus before indexing. On the other hand, if the CLQA system only has access to offline MT, the SMLIR model can take full advantage of having the translated corpus in the index.

3. Methods for Detecting and Correcting MT Errors: In Chapters 5 and 6, we presented two types of adequacy-oriented automatic post-editors (APEs). While the CLIR models addressed the recall errors identified during our error analysis, the goal of the APEs was to address errors in result understanding. Even if a relevant result is retrieved, if the result translation is inadequate, the end user may not recognize it as relevant. The APEs focused on the adequacy errors we identified in our error analyses as being most detrimental to the CLQA task, and that we found to be prevalent across different MT systems. Specifically, we presented novel algorithms for the following problems:

- **Detecting NE Mistranslations**: By exploiting bilingual NER tags and a high-precision NE translation dictionary, the algorithm is able to flag NEs in MT output that may be mistranslated. A related algorithm specifically looks for NEs from the query that are mistranslated in document translations.

- **Detecting Missing Content Words**: By using MT word alignments and bilingual POS tagging, the algorithm is able to identify missing content words that may arise from three different types of MT errors (deletion, content-to-function mistranslation and partial phrase translation).

  These techniques are lightweight, language-independent and MT system-independent algorithms for automatically detecting fine-grained, phrase-level adequacy errors in MT output.

- **Finding Context-Specific Translation Suggestions**: Once an error is flagged, we use the CLQA context to look for a translation correction. The suggestions may come from the query, translations of related documents (or sentences), internal MT resources (such as

SMT phrase dictionaries) or external resources (such as Wikipedia-mined dictionaries). The algorithm also scores and ranks the suggestions for a particular flagged error.

- **Rule-Based and Feedback APEs**: Once phrases in a sentence are flagged as potential errors and a ranked list of translation suggestions is generated for each error, the APEs apply the suggestions to the translated sentence in different ways. The feedback APE does a second-pass decoding with the new translation suggestions, which improves adequacy while retaining fluency. The rule-based APE uses heuristics based on word alignments to re-write the existing MT output. Since it post-edits almost all of the flagged errors, it has a larger positive impact on translation adequacy; however, the post-edited sentences are much less fluent than the feedback post-edited sentences.

4. TLIR Evaluation Corpus: In Chapter 7, we described the evaluation corpus we built for the TLIR task, which had the following properties:

- **Gold Reference Translations**: Unlike CLIR evaluation corpora, the TLIR evaluation corpus has human reference translations, so that machine-translated search results can be compared to gold translations. This allows us to run MT evaluations and automatic metrics, which is typically not possible with standard CLIR corpora.

- **Gold CLIR Judgments**: Unlike MT evaluation corpora, this corpus has been augmented with CLIR queries and annotated query-sentence pairs. The query-sentence pairs were annotated for gold relevance using human reference translations.

- **MT Relevance Judgments**: We crowd-sourced relevance judgments on translated search results for two different MT systems across two different genres.

- **Intrinsic and Extrinsic Metrics**: While TLIR systems can be implemented in a variety of ways, they all must do retrieval and translation (in any order). By using MT output for both the retrieval and result translation steps, we can extrinsically evaluate systems on the entire end-to-end TLIR task. On the other hand, if we use MT for retrieval, but then present search results to the user in human-translated form, we can focus on the impact of MT on CLIR only. (This is akin to having “oracle” or perfect result translation.) Conversely, if we do retrieval over human translation, and then present search results in MT form, we are evaluating the impact of MT on relevance understanding only. Using human translations for both retrieval and relevance annotation provides an upper bound for the TLIR task.

5. Task-Embedded MT Evaluation: Using the TLIR evaluation corpus, we were able to measure the impact of our novel CLIR models and our APEs on the end-to-end TLIR task, as well as isolate their impact on each step in the pipeline, namely result retrieval and translated result understanding. Our main findings were as follows:

- Result translation had a significant impact on the end-to-end TLIR system: using MT instead of reference translations degraded TLIR performance by 5-19% across different experimental conditions. While result translation is essential for real-world applications of CLIR, standard CLIR evaluations do not take it into account. These findings show that the impact of MT on CLIR result translation is large and quantifiable.

- While the HiFST translation of the newswire corpus appeared to be better than the Moses translation intrinsically (by BLEU score), HiFST actually had lower translated relevance than Moses because Moses was better at translating NEs. This shows how specific aspects of MT are crucial for certain tasks, and explains why task-agnostic MT systems need to be adapted to the needs of different applications.

- Similarly, while the QT model performed better than the DT model in the intrinsic CLIR evaluation (with perfect result translation) in the newswire genre, the opposite was true in the TLIR evaluation, where MT was used for both retrieval and result translation. Both of these results show the importance of evaluating models extrinsically as well as intrinsically.

- The adequacy-oriented APEs were successful in addressing “lost in result translation” errors: they improved sentence relevance 4-28% of the time, and rarely degraded relevance. This shows that task-agnostic MT output can be adapted to specific task needs.

- As in the CLIR evaluation, the hybrid SMLIR and QT-rerank models performed better than either QT or DT in the end-to-end TLIR evaluation. The TLIR analysis showed exactly why: the QT model always did better than DT on CLIR with perfect result translation, but DT outperformed QT on translated results understanding (with upper-bound retrieval). The hybrid models tightly integrated QT and DT, resulting in better performance than both QT and DT in the end-to-end TLIR condition.

8.3 Limitations

Our experiments were limited in scope to Chinese-English and Arabic-English language pairs and cross-lingual tasks with NE-focused queries. We chose these experimental settings because they are anchored in real-world needs and they present challenges in both the MT and CLIR aspects of the task. In this section, we discuss the reasons for these experimental design decisions in more detail.

Error Analysis: The conclusions in our error analysis are based on our detailed analysis of the output of three different MT systems, and supported by our experience in handling the output of multiple additional MT systems throughout the course of this thesis. Our experiments were limited to Arabic-English and Chinese-English SMT systems, though error analyses by other researchers have supported these conclusions for other language pairs and other SMT systems. It would also be interesting to see what kinds of adequacy errors occur in rule-based (non-statistical) MT systems, but that is beyond the scope of this thesis.

One limitation we faced while doing the error analyses was that it is very difficult to directly compare MT systems across different years. MT evaluations typically constrain the training data and use a test set from a time period after the end of the training data. Systems built after a particular MT evaluation will have trained on new data, including data from the time period during and after the test set. Also, once an MT evaluation set is released, later MT systems often use it as a tuning or development set, so it can no longer be used as a point of comparison. (For this reason, the NIST OpenMT progress test set has very restrictive conditions limiting human interaction with the data.[1])

Since it was infeasible to directly compare MT systems from different years on the same test data, we chose instead to broaden the scope of the error analyses, focusing on a different aspect of MT for each one, while retaining the CLQA task as the motivation. In the first error analysis, we focused on how MT errors impact CLIR precision and recall, and also explored how different kinds of NE mistranslations arise during MT. In the second error analysis, we quantified different

[1] http://www.itl.nist.gov/iad/mig/tests/mt/2009/MT09_ProgressTestForm.pdf

kinds of adequacy errors across both newswire and web genres. Based on that analysis, we focused the third analysis on missing and deleted content words, which are very harmful to translation adequacy. All of our work in later chapters was directly motivated by the results of these three error analyses.

CLQA Tasks: Throughout the thesis, defining what the CLQA task was remained a tricky question. The CLIR evaluation (in Chapter 4) was a document-level evaluation done in the source language (Chinese) with a small number of trained annotators, but the relevance guidelines were taken from very specific GALE distillation guidelines. When we used similar guidelines in a crowd-sourced evaluation (in Chapter 5) to judge sentence-level relevance in translation, we got inconclusive results because it was difficult to apply such narrow relevance guidelines to garbled MT sentences (without having reference translations as a gold standard). For the TLIR evaluation corpus (in Chapter 7), we broadened the task definition to finding facts about a particular entity. This task falls between sentence-level IR and QA: simply mentioning the query entity in a sentence is neither necessary nor sufficient for relevance, yet the task does not require the kind of deep sentence understanding that is associated with many QA tasks.

We believe that our techniques are generally applicable to many cross-lingual information access applications, including both CLIR and CLQA. Our experiments were limited by the lack of a shared evaluation corpus, which motivated us to create our own. Using crowd-sourced annotators was a fast way to gather a large number of judgments. But it also imposed additional limitations, since annotators were typically untrained English speakers, so we ended up using a broader task definition and using reference translations as the gold standard for relevance rather than source-language judgments, which is typical in CLIR.

Query Types: All of the tasks we evaluated focused on names because NEs play a major role in a variety of tasks. The DARPA GALE evaluation is based on the needs of intelligence analysts who track people and groups. In the official Y2 evaluation, over 90% of the queries had at least one NE in them. In web search, NEs are also important: an estimated 71% of web queries contain a name [Guo *et al.*, 2009].

Not only are NEs crucial for many tasks, but they pose a particular challenge for CLIR and CLQA because NEs are especially difficult for MT systems to translate. Focusing on NEs forced us to go beyond simply using MT to translate the queries and explore multiple techniques and resources for NE translation, such as NE dictionaries mined from Wikipedia and searching in related translated documents. These methods were ultimately useful for finding translations for non-NE terms as well, as we discovered when applying our APEs to deleted content words.

While our experiments focused on NEs, our methods can be adapted to other types of queries. For instance, if the query is a long, open-ended (non-templated) question, then sentence-level translation (via SMT) can be combined with phrase-level translation resources to improve query translation, and additional question analysis can be done in both languages, such as parsing or semantic role labelling, to determine the information need expressed by the question. Aspects of the system can also be adapted to different answer types. For tasks with narrower relevance guidelines, SMLIR can be used for first-pass, high-recall document retrieval, and then a separate sentence selection module can be used to do additional analysis and identify relevant sentences.

Language Pairs: Our CLIR experiments were limited to the Chinese-English and Arabic-English language pairs, and our APE experiments were limited to Arabic-English. These language pairs were had relatively decent quality MT, so a TLIR system would be expected to do quite well on them. However, our error analysis showed poor performance on the TLIR task, and further isolated the cause to several specific types of MT adequacy errors.

Carrying out TLIR experiments on a language pair with very low quality MT would not be as interesting because the upper limit for TLIR performance would still be quite low, due to poor result understanding; whereas a language pair with higher quality MT might present fewer opportunities for improvement. Consider a French-English TLIR task over a newswire corpus. In this case, the MT quality is much higher, so there are fewer errors to detect and correct. Since the source and target language are in the same script, leaving NEs untranslated is often acceptable (at least for person names), which makes translating them easier. Because the languages are related, untranslated words are often understandable (or guessable) by the end-user, which makes result understanding less of a challenge.

In contrast, in our experiments, each aspect of the task was a challenge. CLIR was severely affected by MT errors, so we introduced SMLIR and QT-rerank. Query translation was difficult because the MT systems had difficulty translating NEs, so we developed NE-specific resources. Result understanding was impeded by MT adequacy errors, so we designed adequacy-oriented APEs to address them.

While our approaches were motivated by the language pairs we used, it would not be difficult to implement them for other language pairs. The SMLIR model requires a translated corpus and resources for query translation, which can be very simple (such as using the 1-best MT output) or more complex (such as a cascaded approach with multiple resources and back-offs), depending on the available resources. Depending on the morphology of the different languages, different approaches to segmentation, tokenization and stemming are appropriate, as is typical in IR.

The APEs require bilingual POS tags, MT output with word alignments and resources for looking up translation suggestions, which are similar to those for query translation. When post-editing a morphologically rich target language (such as Russian or Arabic), additional processing could help improve grammatical agreement. One approach would be to use a fluency-oriented post-editor, such as the DepFix APE for Czech [Mareček *et al.*, 2011], which post-edits case endings, verb agreements and other grammatical errors. Another approach would be to integrate morphological knowledge into the adequacy-oriented APE. For example, when listing the translation suggestions, all possible inflections of a given word could be considered and scored. Scoring the different inflections would be difficult: if the APE has detected an error in the sentence, then the inflections of other words in the sentence may be incorrect as well, so basing a score on those may lead to further problems. Alternatively, the APE could exploit deeper bilingual analysis (e.g., dependency parses or semantic role labelling) to determine the correct morphology for a given word or phrase.

Corpus Size: The biggest limitation of our TLIR evaluation corpus is that the size is very small for a typical CLIR task, although prior work on studying CLIR results in translation has focused on even smaller user studies. When we carried out a CLQA evaluation on a large corpus with no reference translations (in Chapter 5), we were unable to get conclusive results. In that evaluation, there was no gold standard for relevance or for translation adequacy. To avoid those problems, we needed a corpus with reference translations, so we were limited to relatively small MT test sets.

Since our experiments were not focused on large-scale document-level CLIR, but rather studying the impact of MT on CLIR with result translation, having a small corpus with gold relevance judgments as well as gold reference translations was more useful for our purposes than a larger corpus without reference translations would have been. Our main conclusions held across a variety of experimental conditions: two genres, two MT systems, plus an alternate version of each MT system. If we had a larger evaluation corpus, we could do a deeper analysis of the impact of automatic post-editing on sentence-level relevance. Our current analysis was limited since the APEs only post-edit a small percentage of sentences returned by CLIR, and only a small percentage of those sentences are actually relevant, so the number of sentences that the APE can potentially improve is small. Future shared tasks (such as the current DARPA BOLT tasks) may make larger corpora available for studying task-embedded MT.

8.4 Future Directions

In this thesis, we took a holistic view of the TLIR task, and adapted the CLIR and MT systems to the needs of the task. The SMLIR and QT-rerank models assumed that search results would have to be translated before being displayed to the user, and exploited this extra information. The APEs utilized resources from the CLIR task to automatically detect MT errors that impact the task and to find better translations for those errors. By tightly integrating the MT and CLIR, we improved both intrinsic measures of MT adequacy and CLIR performance, as well as the end-to-end TLIR system. There are several different threads of future work that could extend the work we have presented so far.

More Powerful APEs: The adequacy-oriented APEs we presented were lightweight, language-independent, general to any MT system and successfully improved MT adequacy. Future APEs could use additional sources of knowledge to improve MT even more. Our APEs used only shallow linguistic knowledge, such as POS tags and NER tags. Using bilingual parses could help the APEs identify errors better, as well as make more informed decisions about how to re-write the MT output. Another direction for APEs is to focus on different aspects of translation besides adequacy. Previous work in APE has focused on grammatical agreement and fluency; future work could focus on discourse-level errors, such as coherence and consistency.

Interactive MT: The feedback APE we presented uses task resources to detect and correct errors. Many online MT systems allow users to send feedback directly to the system by highlighting errors and typing in corrections. How to best incorporate this feedback (and discard bad feedback) back into the MT system is an open research question, which automatic post-editing could help address.

Task-Embedded MT: We believe that looking at MT from a task-oriented perspective can help in other applications besides CLIR and CLQA. For instance, many travel sites use MT to translate user-generated reviews and also independently run sentiment analysis tools to extract phrases with high polarity. By applying the sentiment tools during review translation, the sites could make sure that sentiment and polarity are preserved across the languages (so that a restaurant that is "not so good" does not become "so good.") By using MT, the site could improve sentiment accuracy for languages with few sentiment training resources.

In the future, we believe that MT will continue to be used in increasingly diverse tasks and applications. Whether it is a multilingual team of gamers chatting about attack strategies, aid workers who need instantaneous translations of text messages to respond to a humanitarian crisis, fans watching the latest episode of a tv show with translated subtitles, or teenagers tweeting about a popular rock star's latest concert in a foreign country, each application of MT has different needs. Doing a task-oriented MT error analysis can demonstrate the usefulness of MT in each task, and highlight specific aspects of MT that are most important to each task. Leveraging information from the task context can be used to adapt the output of open domain MT systems to the needs of the task. And tighter integration of MT with other task components can lead to better end-to-end performance.

Appendix A

Template-Based Queries

A.1 GALE Y2 Query Templates

Template-based queries for the DARPA GALE Y2 distillation evaluation. Excerpted from BAE Systems Advanced Information Technologies "Go/NoGo Formal Distillation Evaluation Plan for GALE" (version 2.1, April 13, 2007).

```
1: LIST FACTS ABOUT EVENT [event]
Activity Date: indicates date of the event
Location: indicates the location of the event

2: WHAT [people|organizations|countries] ARE INVOLVED IN [event]
AND WHAT ARE THEIR ROLES?
Activity Date: indicates date of the event
Location: indicates the location of the event

3: PROVIDE INFORMATION ON [person]
Activity Date: not applicable for this template
Location: not applicable for this template
```

4: PROVIDE INFORMATION ON [organization]

Activity Date: not applicable for this template

Location: not applicable for this template

5: FIND STATEMENTS MADE BY OR ATTRIBUTED TO [person] ON [topic(s)]

Activity Date: when provided, responses should only include statements made during the specified time window

Location: not applicable for this template

6: DESCRIBE THE RELATIONSHIP OF [person/org] TO [person/org]

Activity date: if provided, only include activities associated with the relationship (such as meetings, discussions, and major events) that occurred within the specified time interval.

Location: when provided, responses should only include evidence of the relationship (such as meetings, business activity) that occurred in specified location.

7: DESCRIBE INVOLVEMENT OF [person/organization/country] IN [event/topic]

Activity Date: specifies date of the event, or period of interest for the topic

Location: when provided, specifies the location of the event or topic (ex: coal mining in West Virginia)

8: DESCRIBE THE PROSECUTION OF [person] FOR [crime]

Activity Date: indicates time frame for the prosecution

Location: indicates the location of the prosecution

9: HOW DID [country] REACT TO [event]?
Activity Date: indicates date of the event
Location: indicates the location of the event

10: WHAT CONNECTIONS ARE THERE BETWEEN
[event1/topic1] and [event2/topic2]?
Activity Date: undetermined so far
Location: undetermined so far

11: FIND ACQUAINTANCES OF [person]
Activity date: responses restricted to acquaintance ties
that are explicitly declared to exist during the
specified time period
Location: responses restricted to acquaintances that are
explicitly connected to the specific location (ex: friends
and business partners in Puerto Rico)

12: PRODUCE A BIOGRAPHY OF [person]
Activity Date: not applicable for this template
Location: not applicable for this template

13: DESCRIBE THE ACTIONS OF [person] DURING [date] TO [date]
Activity Date: indicates time interval of interest for the query
Location: indicates location of actions of interest for the query

14: DESCRIBE THE ACTIONS OF [organization] DURING [date] TO [date]
Activity Date: indicates time interval of interest for the query
Location: indicates location of actions of interest for the query

15: DESCRIBE ARRESTS OF PERSON FROM [organization]
AND GIVE THEIR ROLE IN ORGANIZATION
Activity Date: indicates time interval of the arrests of interest
Location: indicates location of the arrests

16: DESCRIBE ATTACKS in [location] GIVING LOCATION
(AS SPECIFIC AS POSSIBLE),
DATE, AND NUMBER OF DEAD AND INJURED.
Activity Date: indicates time interval of the attacks to be
included in the response
Location: indicates location of the attacks of interest to for the
query

17: WHERE HAS [person] BEEN AND WHEN?
Activity Date: if specified, only include locations whose known
associated date falls within the time interval
Location: not applicable for this template

A.2 GALE Y4 Query Templates

Template-based queries for the DARPA GALE Y4 distillation evaluation. Excerpted from BAE Systems Advanced Information Technologies “Phase 4 Formal Evaluation Plan For GALE Distillation” (version 1.3, October 19, 2009).

Note that Phase 2 templates 10 and 12 were dropped from Phase 4 evaluation, and templates 18-22 are new templates for Phase 4. The total number of templates in Phase 4 is 20.

1: LIST FACTS ABOUT EVENT [event]

Activity Date: indicates date of the event

Location: indicates the location of the event

2: WHAT [people|organizations|countries] ARE INVOLVED IN [event]?

Activity Date: indicates date of the event

Location: indicates the location of the event

3: PROVIDE INFORMATION ON [person]

Activity Date: not applicable for this template

Location: not applicable for this template

4: PROVIDE INFORMATION ON [organization]

Activity Date: not applicable for this template

Location: not applicable for this template

5: FIND STATEMENTS MADE BY OR ATTRIBUTED TO [person] ON [topic(s)]

Activity Date: when provided, responses should only include statements made during the specified time window

Location: not applicable for this template

6: DESCRIBE THE RELATIONSHIP OF [person/org] TO [person/org]
Activity date: not applicable
Location: not applicable

7: DESCRIBE INVOLVEMENT OF [person/organization/country] IN [event/topic]
Activity Date: specifies date of the event, or period of interest for the topic
Location: when provided, specifies the location of the event or topic (ex: coal mining in West Virginia)

8: DESCRIBE THE PROSECUTION OF [person] FOR [crime]
Activity Date: indicates time frame for the prosecution
Location: indicates the location of the prosecution

9: HOW DID [country] REACT TO [event]?
Activity Date: indicates date of the event
Location: indicates the location of the event

11: FIND ACQUAINTANCES OF [person]
Activity date: not applicable
Location: not applicable

13: DESCRIBE THE ACTIONS OF [person] DURING [date] TO [date]
Activity Date: N/A
Location: indicates location of actions of interest for the query

14: DESCRIBE THE ACTIONS OF [organization] DURING [date] TO [date]
Activity Date: N/A
Location: indicates location of actions of interest for the query

15: DESCRIBE ARRESTS OF PERSON FROM [organization] AND GIVE THEIR ROLE IN ORGANIZATION
Activity Date: indicates time interval of the arrests of interest
Location: indicates location of the arrests

16: DESCRIBE ATTACKS in [location] GIVING LOCATION (AS SPECIFIC AS POSSIBLE), DATE, AND NUMBER OF DEAD AND INJURED.
Activity Date: indicates time interval of the attacks to be included in the response
Location: indicates location of the attacks of interest to for the query

17: WHERE HAS [person] BEEN AND WHEN?
Activity Date: if specified, only include locations whose known associated date falls within the time interval
Location: not applicable for this template

18: LIST LOCATIONS OF REPRESENTATIVES OF [organization/GPE]
Activity dates: Specifies the time interval for the locations of travel and home/office
Location: Not applicable

19: DESCRIBE A MEETING OR CONTACT BETWEEN [person/organization] AND [person/organization]
Activity dates: responses are restricted to meetings that happened within a specified time frame
Location: responses are restricted to meetings where both parties were at the specified location

20: FIND PEOPLE WHO VISITED [location]

Activity dates: responses are restricted to visits that happened within a specified time frame

Location: not applicable

21: DESCRIBE THE ELECTION CAMPAIGN OF [person]

Activity dates: responses are restricted to activities that happened within a specified time frame

Location: responses are restricted to activities that took place in the specified location

22: WHICH SOURCES MADE STATEMENTS ON [topic]

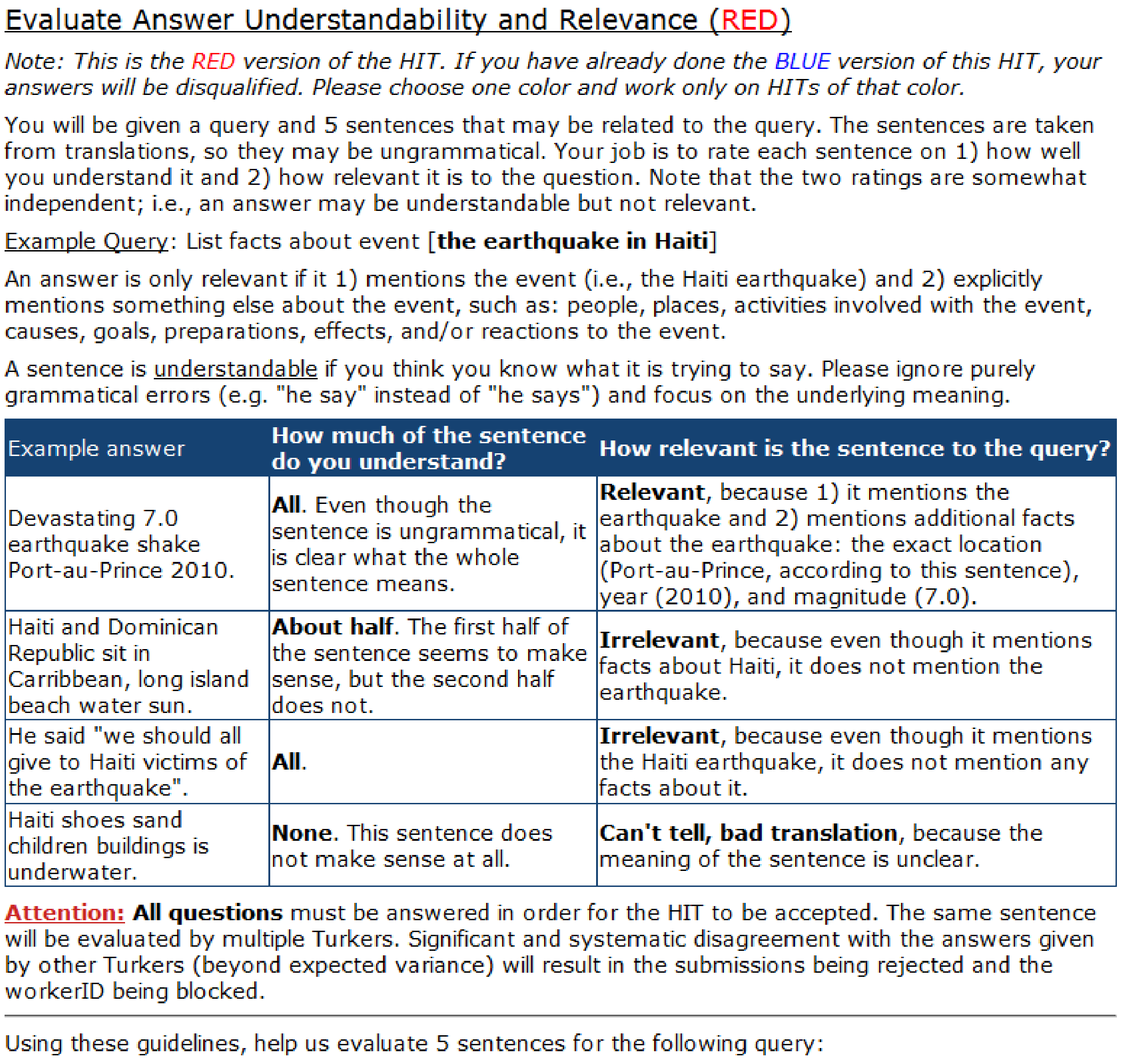
Activity dates: the topic of interest falls within a specified time frame

Location: the topic of interest is restricted to the specified location

Appendix B

Annotation Instructions and Judging Interfaces

B.1 Annotation Interface for Chapter 5



Evaluate Answer Understandability and Relevance (RED)

Note: This is the RED version of the HIT. If you have already done the BLUE version of this HIT, your answers will be disqualified. Please choose one color and work only on HITs of that color.

You will be given a query and 5 sentences that may be related to the query. The sentences are taken from translations, so they may be ungrammatical. Your job is to rate each sentence on 1) how well you understand it and 2) how relevant it is to the question. Note that the two ratings are somewhat independent; i.e., an answer may be understandable but not relevant.

Example Query: List facts about event [**the earthquake in Haiti**]

An answer is only relevant if it 1) mentions the event (i.e., the Haiti earthquake) and 2) explicitly mentions something else about the event, such as: people, places, activities involved with the event, causes, goals, preparations, effects, and/or reactions to the event.

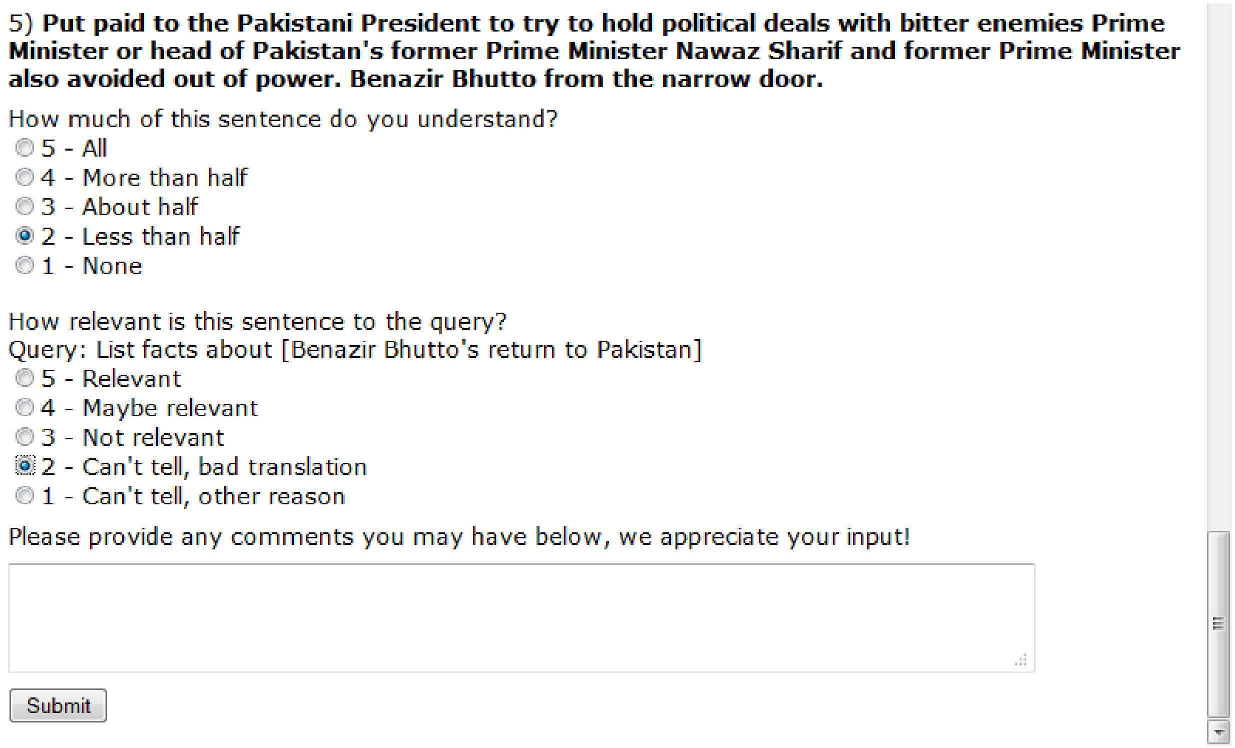
A sentence is understandable if you think you know what it is trying to say. Please ignore purely grammatical errors (e.g. "he say" instead of "he says") and focus on the underlying meaning.

Example answer	How much of the sentence do you understand?	How relevant is the sentence to the query?
Devastating 7.0 earthquake shake Port-au-Prince 2010.	**All**. Even though the sentence is ungrammatical, it is clear what the whole sentence means.	**Relevant**, because 1) it mentions the earthquake and 2) mentions additional facts about the earthquake: the exact location (Port-au-Prince, according to this sentence), year (2010), and magnitude (7.0).
Haiti and Dominican Republic sit in Carribbean, long island beach water sun.	**About half**. The first half of the sentence seems to make sense, but the second half does not.	**Irrelevant**, because even though it mentions facts about Haiti, it does not mention the earthquake.
He said "we should all give to Haiti victims of the earthquake".	**All**.	**Irrelevant**, because even though it mentions the Haiti earthquake, it does not mention any facts about it.
Haiti shoes sand children buildings is underwater.	**None**. This sentence does not make sense at all.	**Can't tell, bad translation**, because the meaning of the sentence is unclear.

Attention: **All questions** must be answered in order for the HIT to be accepted. The same sentence will be evaluated by multiple Turkers. Significant and systematic disagreement with the answers given by other Turkers (beyond expected variance) will result in the submissions being rejected and the workerID being blocked.

Using these guidelines, help us evaluate 5 sentences for the following query:

Figure B.1: Instructions for judging MT understandability and CLQA relevance.



5) **Put paid to the Pakistani President to try to hold political deals with bitter enemies Prime Minister or head of Pakistan's former Prime Minister Nawaz Sharif and former Prime Minister also avoided out of power. Benazir Bhutto from the narrow door.**

How much of this sentence do you understand?

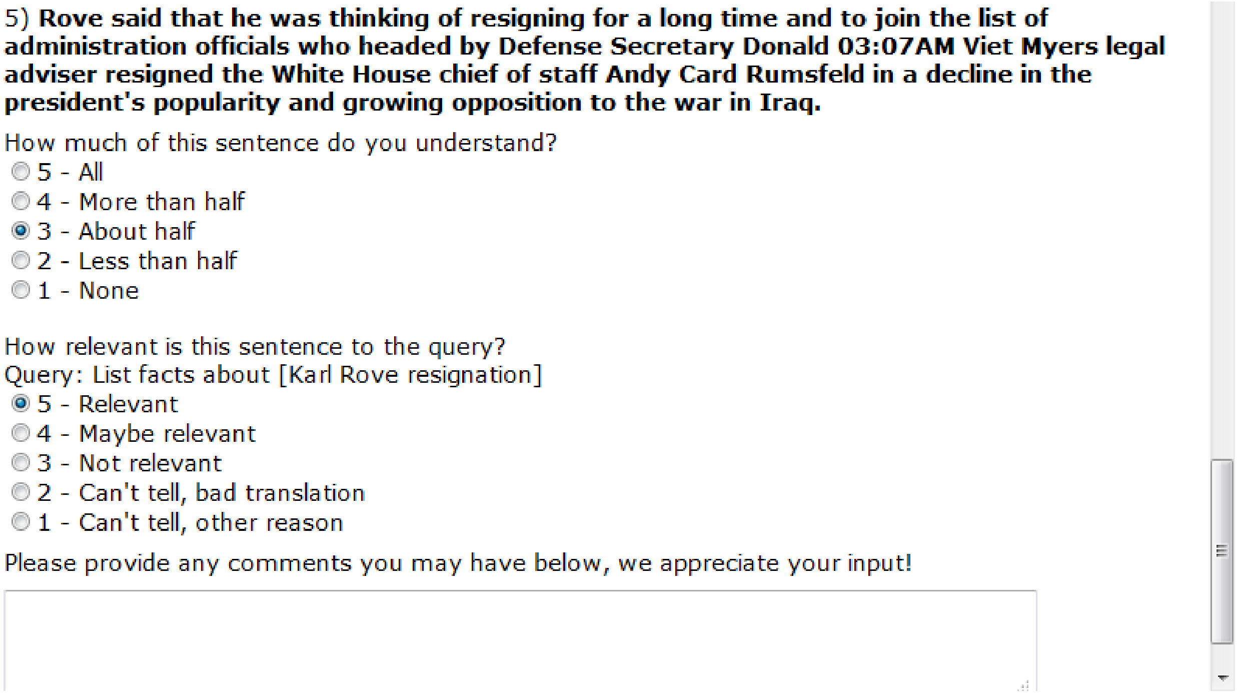
○ 5 - All
○ 4 - More than half
○ 3 - About half
● 2 - Less than half
○ 1 - None

How relevant is this sentence to the query?
Query: List facts about [Benazir Bhutto's return to Pakistan]

○ 5 - Relevant
○ 4 - Maybe relevant
○ 3 - Not relevant
● 2 - Can't tell, bad translation
○ 1 - Can't tell, other reason

Please provide any comments you may have below, we appreciate your input!

Figure B.2: This sentence comes from the better MT system (DTM2-research), but is still too garbled to understand. Though the sentence mentions Bhutto, it is unclear whether it is about her return to Pakistan.



5) **Rove said that he was thinking of resigning for a long time and to join the list of administration officials who headed by Defense Secretary Donald 03:07AM Viet Myers legal adviser resigned the White House chief of staff Andy Card Rumsfeld in a decline in the president's popularity and growing opposition to the war in Iraq.**

How much of this sentence do you understand?

○ 5 - All
○ 4 - More than half
● 3 - About half
○ 2 - Less than half
○ 1 - None

How relevant is this sentence to the query?
Query: List facts about [Karl Rove resignation]

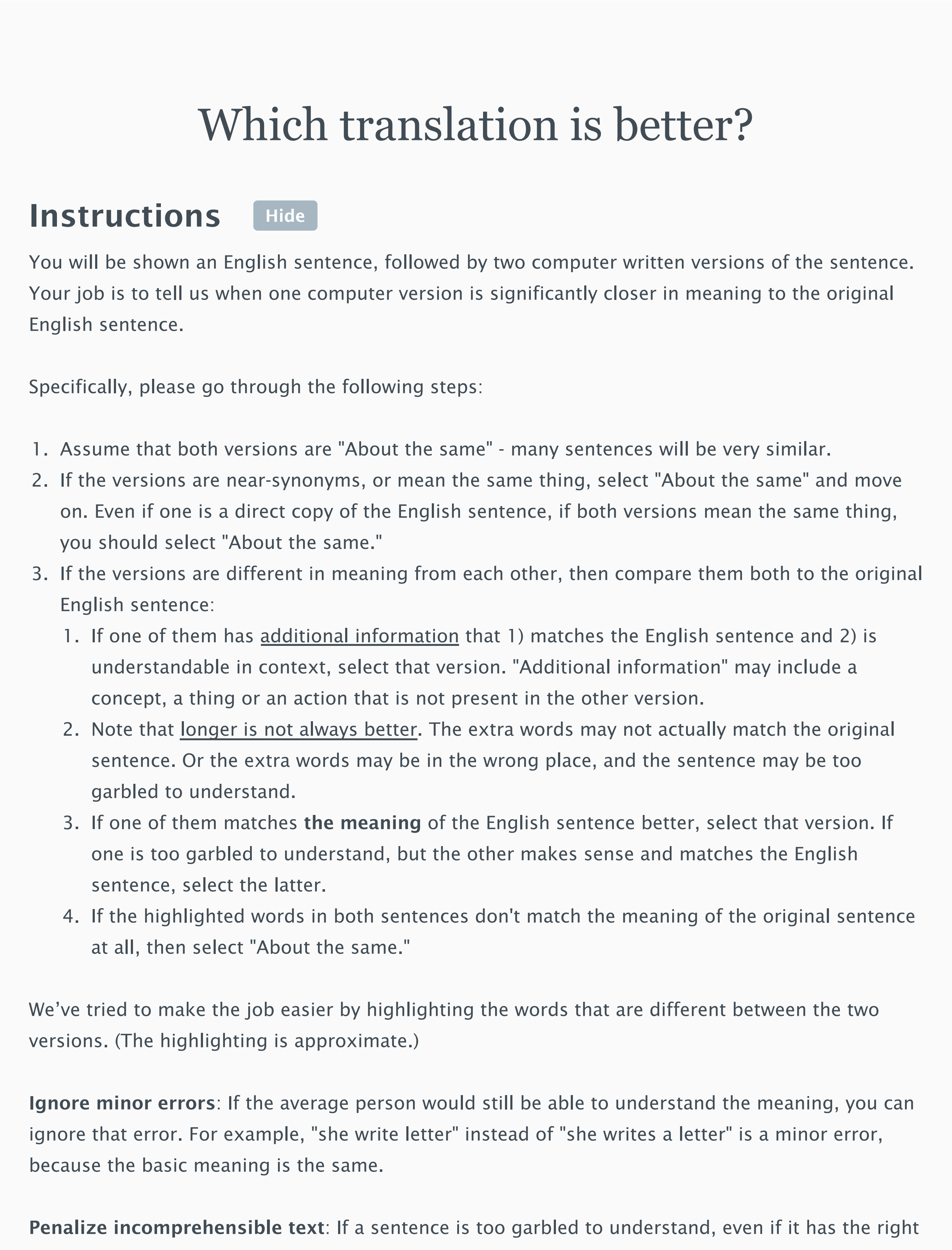
● 5 - Relevant
○ 4 - Maybe relevant
○ 3 - Not relevant
○ 2 - Can't tell, bad translation
○ 1 - Can't tell, other reason

Please provide any comments you may have below, we appreciate your input!

Figure B.3: This sentence is clearly about Rove's resignation, so it is relevant to the query. However, the rest of the sentence is difficult to understand (e.g., "03:07AM Viet Myers"?). This sentence is also from the better MT system (DTM2-research).

B.2 Annotation Interfaces for Chapter 6

Figure B.4: Instructions for judging MT adequacy on CrowdFlower. (Next two pages.)



Which translation is better?

Instructions

You will be shown an English sentence, followed by two computer written versions of the sentence. Your job is to tell us when one computer version is significantly closer in meaning to the original English sentence.

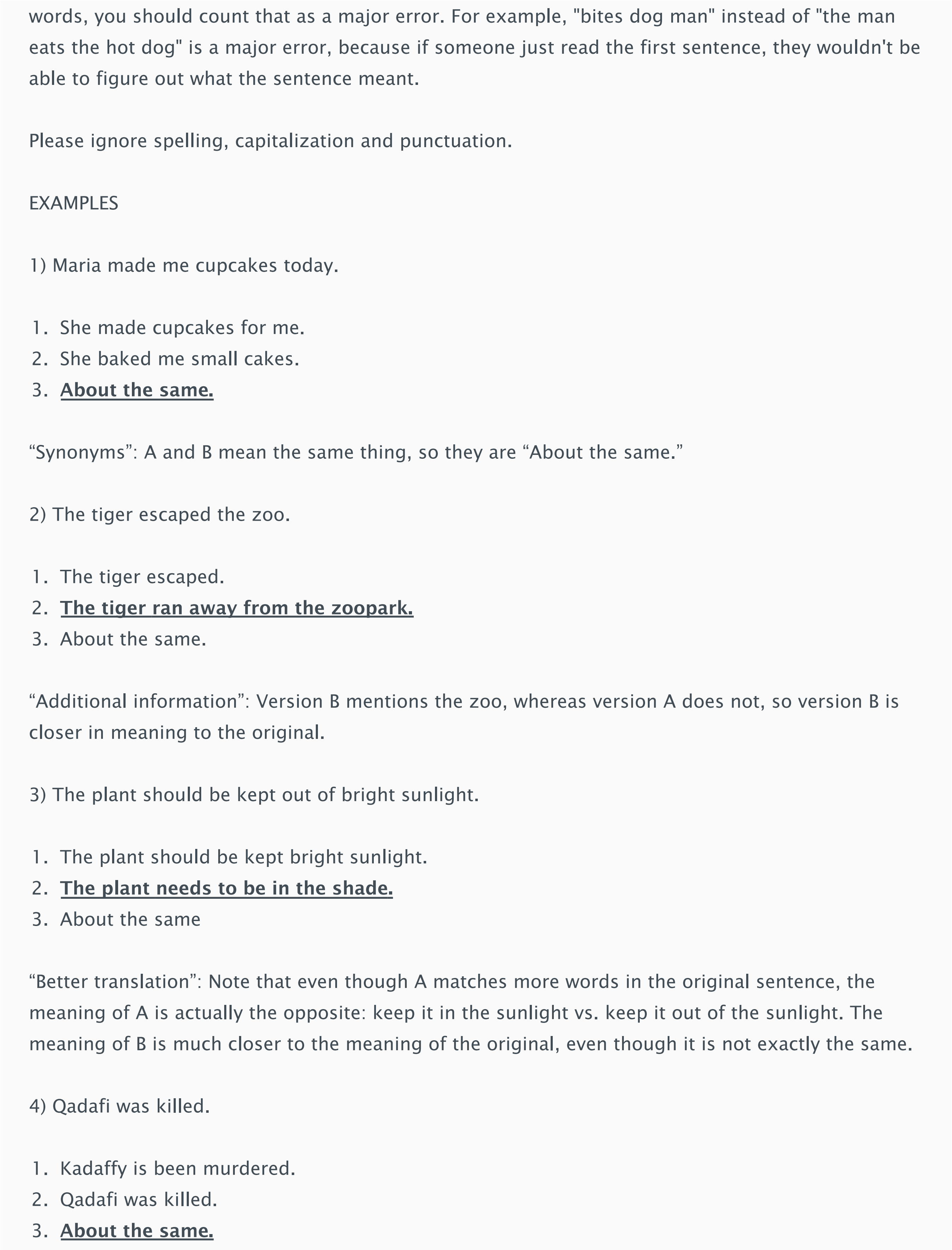
Specifically, please go through the following steps:

1. Assume that both versions are "About the same" - many sentences will be very similar.
2. If the versions are near-synonyms, or mean the same thing, select "About the same" and move on. Even if one is a direct copy of the English sentence, if both versions mean the same thing, you should select "About the same."
3. If the versions are different in meaning from each other, then compare them both to the original English sentence:
   1. If one of them has additional information that 1) matches the English sentence and 2) is understandable in context, select that version. "Additional information" may include a concept, a thing or an action that is not present in the other version.
   2. Note that longer is not always better. The extra words may not actually match the original sentence. Or the extra words may be in the wrong place, and the sentence may be too garbled to understand.
   3. If one of them matches **the meaning** of the English sentence better, select that version. If one is too garbled to understand, but the other makes sense and matches the English sentence, select the latter.
   4. If the highlighted words in both sentences don't match the meaning of the original sentence at all, then select "About the same."

We’ve tried to make the job easier by highlighting the words that are different between the two versions. (The highlighting is approximate.)

Ignore minor errors: If the average person would still be able to understand the meaning, you can ignore that error. For example, "she write letter" instead of "she writes a letter" is a minor error, because the basic meaning is the same.

Penalize incomprehensible text: If a sentence is too garbled to understand, even if it has the right



words, you should count that as a major error. For example, "bites dog man" instead of "the man eats the hot dog" is a major error, because if someone just read the first sentence, they wouldn't be able to figure out what the sentence meant.

Please ignore spelling, capitalization and punctuation.

EXAMPLES

1) Maria made me cupcakes today.

1. She made cupcakes for me.
2. She baked me small cakes.
3. **About the same.**

“Synonyms”: A and B mean the same thing, so they are “About the same.”

2) The tiger escaped the zoo.

1. The tiger escaped.
2. **The tiger ran away from the zoopark.**
3. About the same.

“Additional information”: Version B mentions the zoo, whereas version A does not, so version B is closer in meaning to the original.

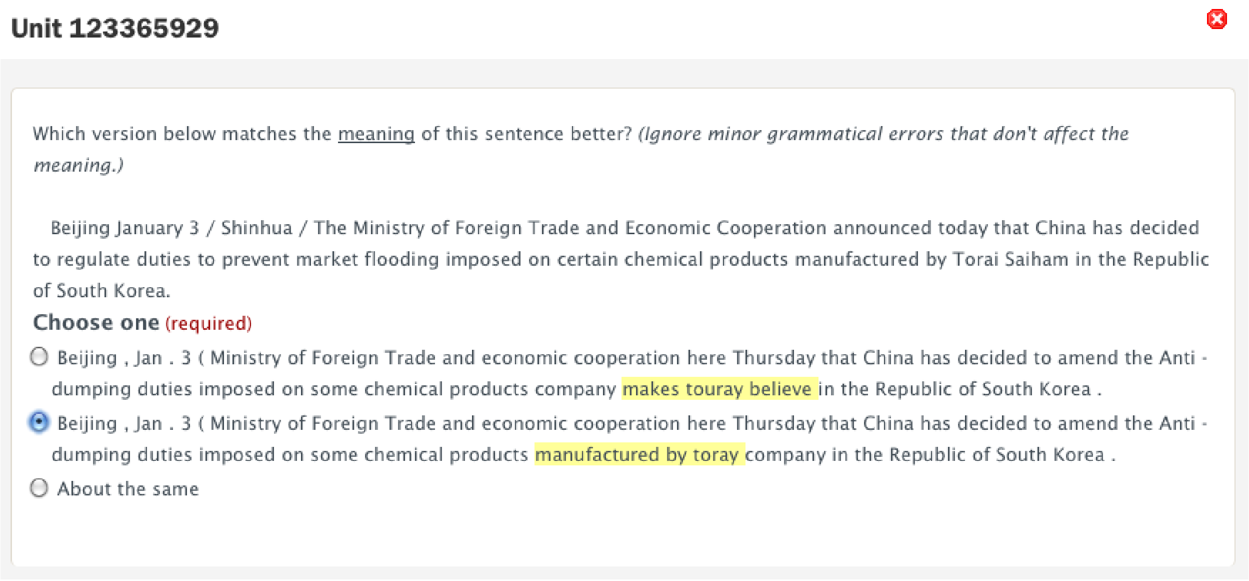
3) The plant should be kept out of bright sunlight.

1. The plant should be kept bright sunlight.
2. **The plant needs to be in the shade.**
3. About the same

“Better translation”: Note that even though A matches more words in the original sentence, the meaning of A is actually the opposite: keep it in the sunlight vs. keep it out of the sunlight. The meaning of B is much closer to the meaning of the original, even though it is not exactly the same.

4) Qadafi was killed.

1. Kadaffy is been murdered.
2. Qadafi was killed.
3. **About the same.**



Unit 123365929

Which version below matches the meaning of this sentence better? *(Ignore minor grammatical errors that don't affect the meaning.)*

Beijing January 3 / Shinhua / The Ministry of Foreign Trade and Economic Cooperation announced today that China has decided to regulate duties to prevent market flooding imposed on certain chemical products manufactured by Torai Saiham in the Republic of South Korea.

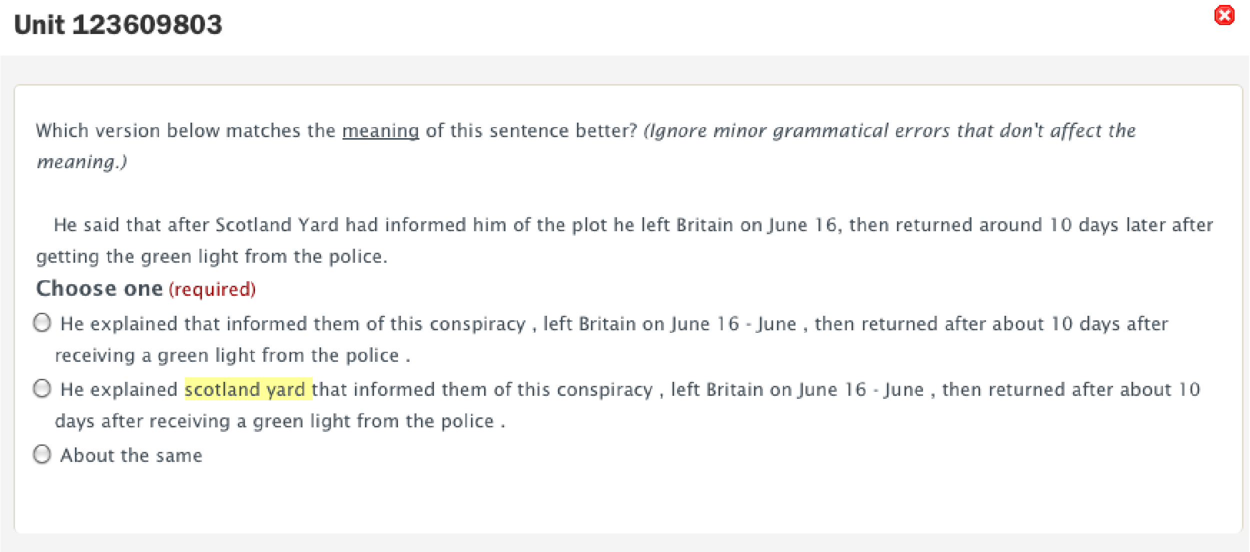
Choose one (required)

○ Beijing , Jan . 3 (Ministry of Foreign Trade and economic cooperation here Thursday that China has decided to amend the Anti - dumping duties imposed on some chemical products company makes touray believe in the Republic of South Korea .

● Beijing , Jan . 3 (Ministry of Foreign Trade and economic cooperation here Thursday that China has decided to amend the Anti - dumping duties imposed on some chemical products manufactured by toray company in the Republic of South Korea .

○ About the same

Figure B.5: Example where the baseline MT is more adequate than the post-edited MT. The APE tried to correct a deleted NE "Saiham," but the suggestion generator did not find the correct translation. Instead, the mistranslation "believe" was inserted, which garbled the neighboring words.



Unit 123609803

Which version below matches the meaning of this sentence better? *(Ignore minor grammatical errors that don't affect the meaning.)*

He said that after Scotland Yard had informed him of the plot he left Britain on June 16, then returned around 10 days later after getting the green light from the police.

Choose one (required)

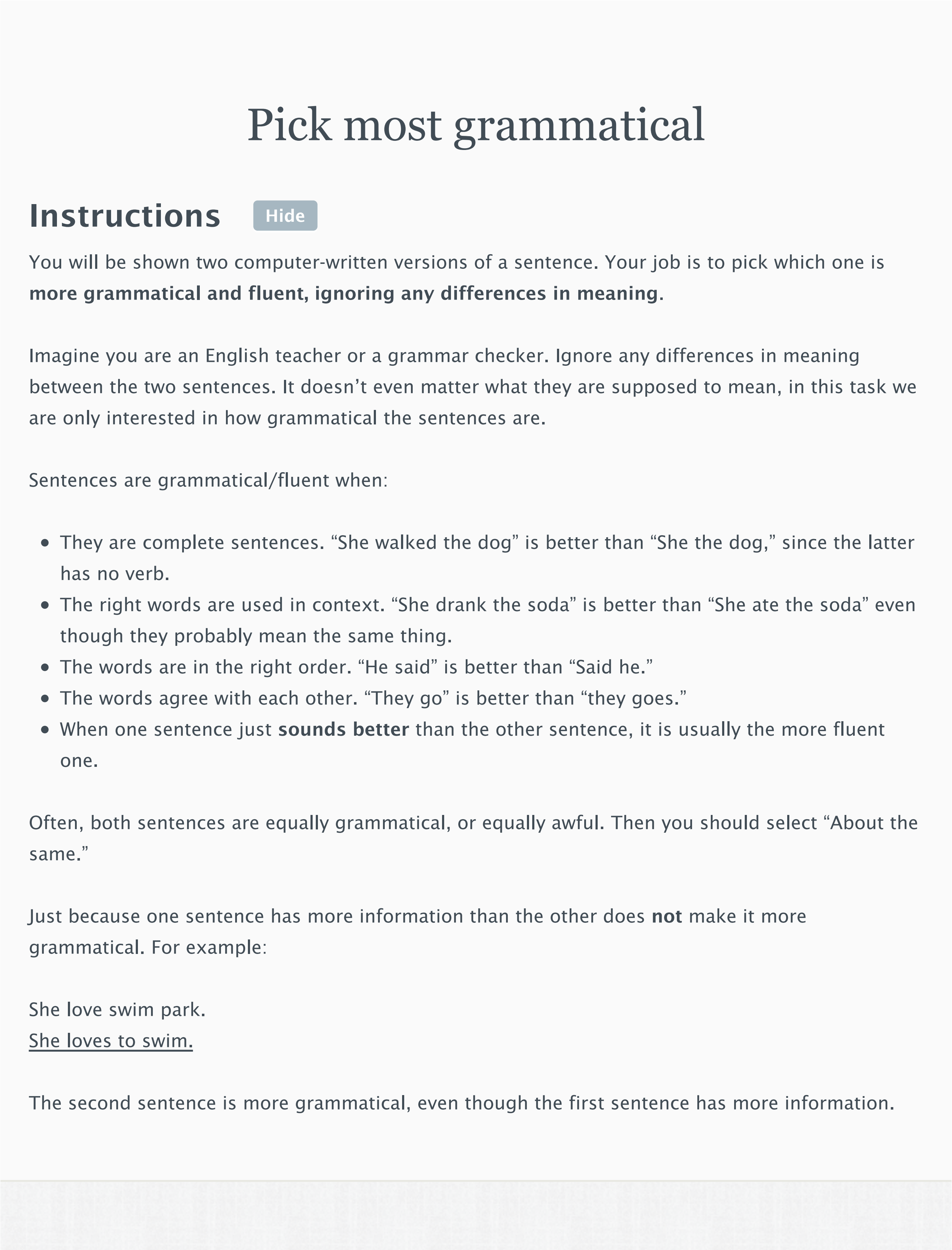
○ He explained that informed them of this conspiracy , left Britain on June 16 - June , then returned after about 10 days after receiving a green light from the police .

○ He explained scotland yard that informed them of this conspiracy , left Britain on June 16 - June , then returned after about 10 days after receiving a green light from the police .

○ About the same

Figure B.6: Example where the post-edited MT is more adequate than the baseline. The rule-based APE correctly inserts the deleted NE "Scotland Yard" and despite the ungrammatical context, it is somewhat understandable.

Figure B.7: Instructions for judging MT fluency on CrowdFlower. (Next page.)



Pick most grammatical

Instructions

You will be shown two computer-written versions of a sentence. Your job is to pick which one is **more grammatical and fluent, ignoring any differences in meaning**.

Imagine you are an English teacher or a grammar checker. Ignore any differences in meaning between the two sentences. It doesn't even matter what they are supposed to mean, in this task we are only interested in how grammatical the sentences are.

Sentences are grammatical/fluent when:

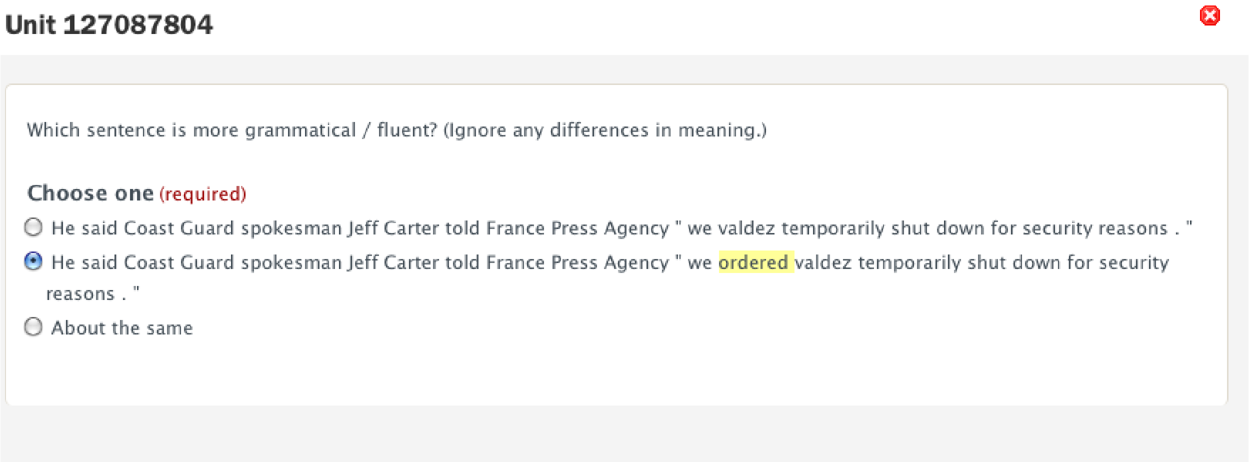
- They are complete sentences. "She walked the dog" is better than "She the dog," since the latter has no verb.
- The right words are used in context. "She drank the soda" is better than "She ate the soda" even though they probably mean the same thing.
- The words are in the right order. "He said" is better than "Said he."
- The words agree with each other. "They go" is better than "they goes."
- When one sentence just **sounds better** than the other sentence, it is usually the more fluent one.

Often, both sentences are equally grammatical, or equally awful. Then you should select "About the same."

Just because one sentence has more information than the other does **not** make it more grammatical. For example:

She love swim park.
She loves to swim.

The second sentence is more grammatical, even though the first sentence has more information.



Unit 127087804

Which sentence is more grammatical / fluent? (Ignore any differences in meaning.)

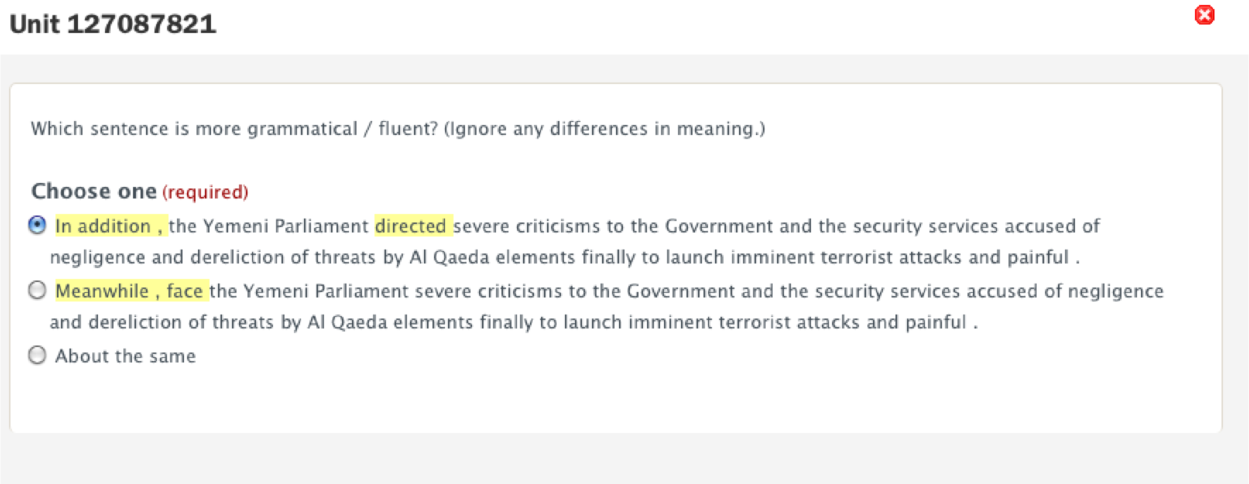
Choose one (required)

○ He said Coast Guard spokesman Jeff Carter told France Press Agency " we valdez temporarily shut down for security reasons . "

● He said Coast Guard spokesman Jeff Carter told France Press Agency " we ordered valdez temporarily shut down for security reasons . "

○ About the same

Figure B.8: The second sentence is more fluent than the first because it retains SVO word ordering ("we ordered valdez shut down") instead of SOV ("we valdez shut down").



Unit 127087821

Which sentence is more grammatical / fluent? (Ignore any differences in meaning.)

Choose one (required)

● In addition , the Yemeni Parliament directed severe criticisms to the Government and the security services accused of negligence and dereliction of threats by Al Qaeda elements finally to launch imminent terrorist attacks and painful .

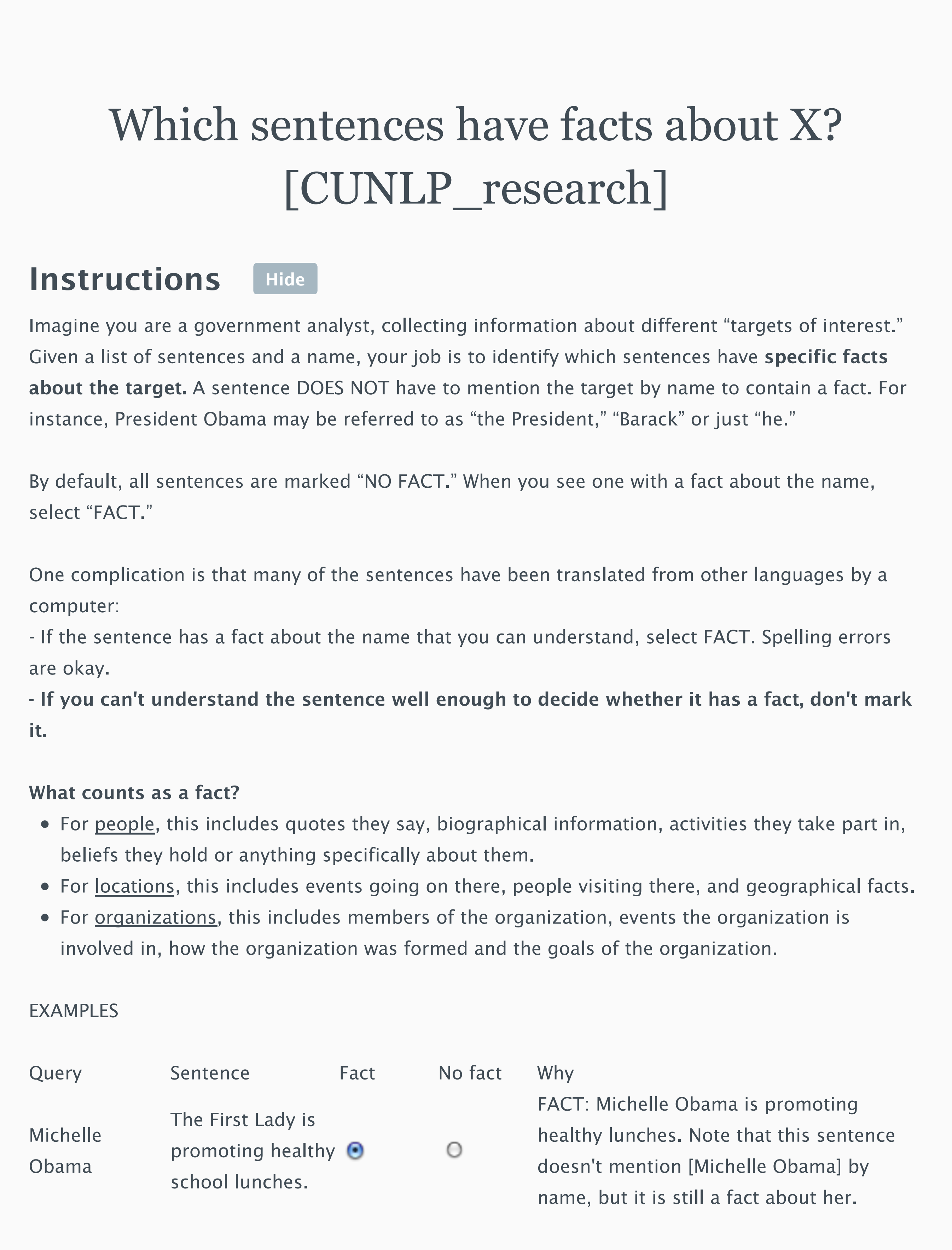
○ Meanwhile , face the Yemeni Parliament severe criticisms to the Government and the security services accused of negligence and dereliction of threats by Al Qaeda elements finally to launch imminent terrorist attacks and painful .

○ About the same

Figure B.9: The first sentence is more fluent because it has SVO word ordering ("Parliament directed criticisms") instead of VSO ("face Parliament criticisms").

B.3 Annotation Interfaces for Chapter 7

Figure B.10: Instructions for judging CLQA relevance on CrowdFlower. (Next two pages.)



Which sentences have facts about X? [CUNLP_research]

Instructions

Imagine you are a government analyst, collecting information about different “targets of interest.” Given a list of sentences and a name, your job is to identify which sentences have **specific facts about the target.** A sentence DOES NOT have to mention the target by name to contain a fact. For instance, President Obama may be referred to as “the President,” “Barack” or just “he.”

By default, all sentences are marked “NO FACT.” When you see one with a fact about the name, select “FACT.”

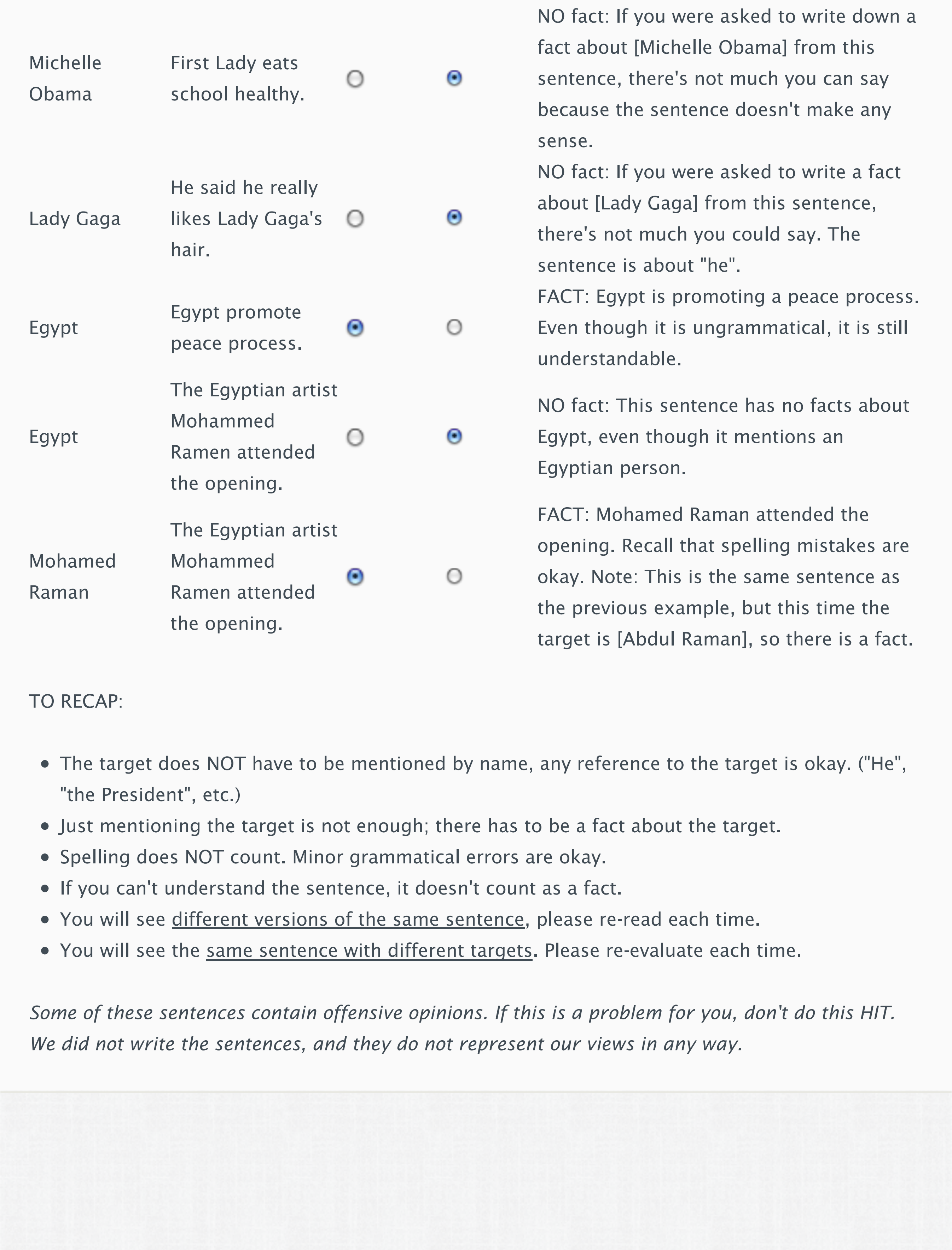
One complication is that many of the sentences have been translated from other languages by a computer:
- If the sentence has a fact about the name that you can understand, select FACT. Spelling errors are okay.
- If you can't understand the sentence well enough to decide whether it has a fact, don't mark it.

What counts as a fact?

- For people, this includes quotes they say, biographical information, activities they take part in, beliefs they hold or anything specifically about them.
- For locations, this includes events going on there, people visiting there, and geographical facts.
- For organizations, this includes members of the organization, events the organization is involved in, how the organization was formed and the goals of the organization.

EXAMPLES

Query	Sentence	Fact	No fact	Why
Michelle Obama	The First Lady is promoting healthy school lunches.	●	○	FACT: Michelle Obama is promoting healthy lunches. Note that this sentence doesn't mention [Michelle Obama] by name, but it is still a fact about her.

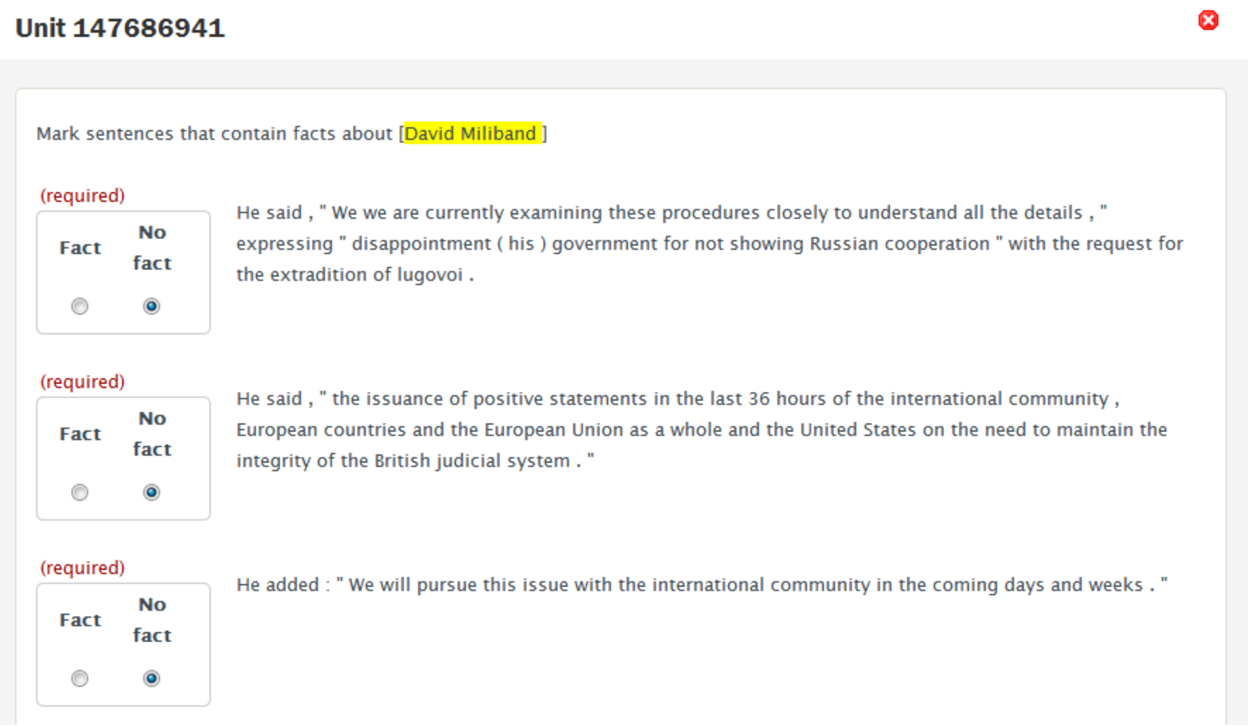


Michelle Obama	First Lady eats school healthy.	○	●	NO fact: If you were asked to write down a fact about [Michelle Obama] from this sentence, there's not much you can say because the sentence doesn't make any sense.
Lady Gaga	He said he really likes Lady Gaga's hair.	○	●	NO fact: If you were asked to write a fact about [Lady Gaga] from this sentence, there's not much you could say. The sentence is about "he".
Egypt	Egypt promote peace process.	●	○	FACT: Egypt is promoting a peace process. Even though it is ungrammatical, it is still understandable.
Egypt	The Egyptian artist Mohammed Ramen attended the opening.	○	●	NO fact: This sentence has no facts about Egypt, even though it mentions an Egyptian person.
Mohamed Raman	The Egyptian artist Mohammed Ramen attended the opening.	●	○	FACT: Mohamed Raman attended the opening. Recall that spelling mistakes are okay. Note: This is the same sentence as the previous example, but this time the target is [Abdul Raman], so there is a fact.

TO RECAP:

- The target does NOT have to be mentioned by name, any reference to the target is okay. ("He", "the President", etc.)
- Just mentioning the target is not enough; there has to be a fact about the target.
- Spelling does NOT count. Minor grammatical errors are okay.
- If you can't understand the sentence, it doesn't count as a fact.
- You will see different versions of the same sentence, please re-read each time.
- You will see the same sentence with different targets. Please re-evaluate each time.

Some of these sentences contain offensive opinions. If this is a problem for you, don't do this HIT. We did not write the sentences, and they do not represent our views in any way.



Unit 147686941

Mark sentences that contain facts about [David Miliband]

(required)

Fact ○ No fact ●

He said , " We we are currently examining these procedures closely to understand all the details , " expressing " disappointment (his) government for not showing Russian cooperation " with the request for the extradition of lugovoi .

(required)

Fact ○ No fact ●

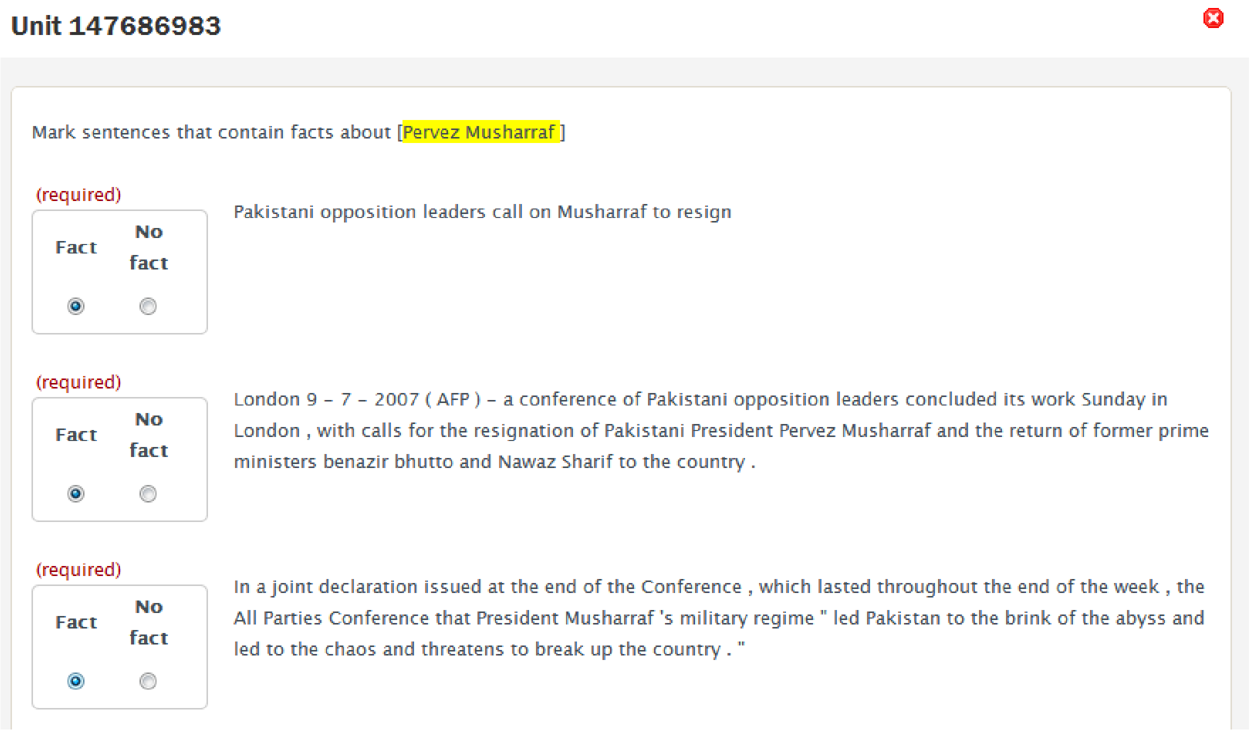
He said , " the issuance of positive statements in the last 36 hours of the international community , European countries and the European Union as a whole and the United States on the need to maintain the integrity of the British judicial system . "

(required)

Fact ○ No fact ●

He added : " We will pursue this issue with the international community in the coming days and weeks . "

Figure B.11: In these MT sentences, despite the three sentences of context, there is no way to tell whether the sentences refer to Miliband, so they are all marked “No fact.” The reference translations do refer to Miliband by name.



Unit 147686983

Mark sentences that contain facts about [Pervez Musharraf]

(required)

Fact ● No fact ○

Pakistani opposition leaders call on Musharraf to resign

(required)

Fact ● No fact ○

London 9 – 7 – 2007 (AFP) – a conference of Pakistani opposition leaders concluded its work Sunday in London , with calls for the resignation of Pakistani President Pervez Musharraf and the return of former prime ministers benazir bhutto and Nawaz Sharif to the country .

(required)

Fact ● No fact ○

In a joint declaration issued at the end of the Conference , which lasted throughout the end of the week , the All Parties Conference that President Musharraf 's military regime " led Pakistan to the brink of the abyss and led to the chaos and threatens to break up the country . "

Figure B.12: This example is also from MT, but here all the sentences clearly have facts about Musharraf.

Glossary

adequacy An adequate translation is one that carries the same meaning as the original text. In other words, adequacy is a measure of how faithful the translation is to the original. In translation theory, adequacy is also referred to as fidelity or faithfulness. Adequacy must be evaluated with respect to the original meaning of the text as well as the translation output, so it requires either a bilingual evaluator or a reference translation.

automatic post-editor (APE) APEs seek to perform the same task as human post-editors: correcting errors in translations produced by MT systems.

BLEU (bilingual evaluation understudy) The de facto standard metric for automatic MT evaluation. The score is based on comparing the MT to one or more reference translations. BLEU calculates the modified n-gram precision for n=1 to N (usually N=4), and then takes the geometric mean, and finally multiplies by a brevity penalty to discourage cheating [Papineni *et al.*, 2002b].

cross-lingual information access (CLIA) This is a broad term that encompasses any cross-lingual task that allows users to search, explore and/or use information in other languages. CLIA tasks may go beyond just CLIR to include additional post-processing steps such as result translation and/or summarization. CLIA also covers a variety of domains, such as cross-lingual image search and cross-lingual patent search.

cross-lingual information retrieval (CLIR) The goal of a CLIR system is to satisfy an information need (expressed as a query) in the query language by returning responses drawn from multilingual corpora, where the corpora are in one or more document language(s), and at least one document language is different from the query language. CLIR differs from TLIR in that it does not require result translation back into the query language; instead, result relevance is evaluated in the document language.

cross-lingual question answering (CLQA) The goal of a CLQA system is to satisfy an information need (expressed as a question) in the query language by returning responses drawn from multilingual corpora, where the corpora are in one or more document language(s), and at least one document language is different from the query language. CLQA differs from TLQA in that it does not require result translation back into the query language; instead, result relevance is evaluated in the document language.

document language In CLIR, the language of the indexed corpus is the document language, which differs from the query language.

document translation (DT) The DT approach to CLIR involves first translating the documents into the query language prior to indexing, and then, at query time, doing a monolingual search in the query language.

fluency A fluent translation is one that is grammatical and understandable by a native speaker of the target language. Fluency is sometimes referred to as transparency, and fluent translations can also be called idiomatic. Fluency is evaluated with respect to the target-language translation only, and is independent of the source-language text.

information need A concept that encompasses exactly what the user is searching for, and determines which of the responses are correct or relevant. The user's query or question is often an imperfect expression of the information need.

Mean average precision (MAP) A standard IR metric that takes into account both recall and precision. It summarizes the overall performance of the system by taking the mean over all queries of the average precision across all levels of recall. Defined in detail in Section 2.2.3.

METEOR (metric for evaluation of translation with explicit ordering) A popular metric for automatic MT evaluation. The score is based on comparing the MT to one or more reference translations. METEOR takes into account both recall and precision, and also has implements flexible string matching based on paraphrase, stemming and synonyms [Lavie and Agarwal, 2007].

machine translation (MT) A machine translation system takes input in the source language and automatically produces output in the target language, with the goal of producing output that is both adequate and fluent.

named entity (NE) A word or phrase that refers to a proper name. The exact definition of NE depends on the named entity recognizer (NER) used and the number/type of classes defined. A simple NER may cover only persons, locations and organizations, while a more complex one may also encompass dates, time, money and miscellaneous.

Normalized Discounted Cumulative Gain (NDCG) A popular IR metric that takes into account the relative ranking that each system gives to the returned documents. Defined in detail in Section 2.2.3.

out of vocabulary (OOV) In MT, an OOV word is one that was not seen in the training data, so the MT system has no knowledge of how to translate it.

part of speech (POS) tags A POS tag classifies a word into a specific word class or lexical category (such as noun, verb, etc.). The number and type of parts of speech depends on the language and POS tagset used.

precision In IR, the fraction of retrieved results that are relevant. In IR evaluations, precision is commonly reported out of the top-k ranked results, for instance, as precision at 10. In MT, precision refers to the number of words or phrases in an MT sentence that match one or more reference translations.

query translation (QT) The QT approach to CLIR involves indexing the documents in the original document language and then, at query time, translating the query into the document language and doing a monolingual search in the document language.

query translation re-rank (QT-rerank) CLIR model introduced in Chapter 4.

query language In CLIR, the user's query is written in the query language, which differs from the document language.

rule-based MT MT systems based on manually written translation rules and dictionaries. RBMT systems are in direct contrast to SMT systems, which acquire translation rules automatically through machine learning over large training corpora, although recently several hybrid SMT-RBMT systems have emerged.

recall In IR, the fraction of relevant results that are retrieved. In MT, recall refers to the number of words or phrases in a reference translation that are expressed in an MT sentence. MT recall roughly corresponds to adequacy.

reference translation A translation written by a human translator. Reference translations are often used in manual MT evaluations so that monolingual annotators can be used to compare MT output to reference translations, rather than having bilingual annotators compare MT output to the source-language text. Reference translations are also frequently used by MT evaluation metrics (such as BLEU), where the MT output is scored by how closely it resembles one or more reference translations.

relevance A search response that satisfies an information need is relevant to that need.

Simultaneous Multilingual Information Retrieval (SMLIR) CLIR model introduced in Chapter 4.

statistical machine translation (SMT) MT systems based on machine learning techniques that learn to translate from one language to another using training data. SMT systems are in direct contrast to RBMT systems, which typically contain hand-built translation rules, although recently several hybrid SMT-RBMT systems have emerged.

source language The language from which an MT system translates; an MT system translates from the source language into the target language.

target language The language into which an MT system translates; an MT system translates from the source language into the target language.

task-embedded MT Any multi-lingual or cross-lingual natural language processing task that is evaluated on MT output.

template-based CLQA A CLQA task where the questions are based on pre-defined templates with argument slots, rather than being open-ended or factoid questions.

TERp (translation edit rate plus) A popular metric for automatic MT evaluation. The "edit rate" between an MT sentence and a set of reference translations is similar to edit distance. TERp has multiple methods for flexible string matching including paraphrase, stemming and synonyms [Snover *et al.*, 2009].

translingual information retrieval (TLIR) Given a query in language ℓ, and a corpus in language m, return relevant results from the corpus, translated from the document language m into the query language. In contrast to CLIR, the TLIR task requires search results to be translated back to the query language.

word or phrase alignments Word (or phrase) alignments represent the mapping from source-language input tokens to target-language output tokens that was used by the decoder to produce the translation. Depending on the model used, the alignments may be word-to-word, phrase-to-phrase or more complex alignments based on translation rules.

Bibliography

[Albrecht and Hwa, 2007] Joshua Albrecht and Rebecca Hwa. A Re-examination of Machine Learning Approaches for Sentence-Level MT Evaluation. In *Proceedings of ACL*, 2007.

[Alonso and Baeza-Yates, 2011] Omar Alonso and Ricardo Baeza-Yates. Design and implementation of relevance assessments using crowdsourcing. In Paul Clough, Colum Foley, Cathal Gurrin, Gareth Jones, Wessel Kraaij, Hyowon Lee, and Vanessa Mudoch, editors, *Advances in Information Retrieval*, volume 6611 of *Lecture Notes in Computer Science*, pages 153–164. Springer Berlin / Heidelberg, 2011.

[Bender *et al.*, 2007] O. Bender, E. Matusov, S. Hahn, S. Hasan, S. Khadivi, and H. Ney. The rwth arabic-to-english spoken language translation system. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU) 2007*, page in press, Kyoto, Japan, December 2007.

[Boschee *et al.*, 2010] Elizabeth Boschee, Marjorie Freedman, Roger Bock, John Graettinger, and Ralph Weischedel. Error analysis and future directions for distillation. In *GALE book (in preparation)*, 2010.

[Brants *et al.*, 2007] Thorsten Brants, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean. Large language models in machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 858–867, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[Brown *et al.*, 1993] Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311, June 1993.

[Buckwalter, 2004] Tim Buckwalter. Buckwalter arabic morphological analyzer version 2.0. *Linguistic Data Consortium (LDC) catalog number LDC2004L02, ISBN 1-58563-324-0*, 2004.

[Callison-Burch *et al.*, 2006] Chris Callison-Burch, Miles Osborne, and Philipp Koehn. Re-evaluating the role of bleu in machine translation research. In *EACL '06: European Chapter of the Association for Computational Linguistics*, pages 249–256, 2006.

[Callison-Burch *et al.*, 2007] Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. (meta-) evaluation of machine translation. In *StatMT '07: Proceedings of the Second Workshop on Statistical Machine Translation*, pages 136–158, Morristown, NJ, USA, 2007. Association for Computational Linguistics.

[Callison-Burch *et al.*, 2012] Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada, June 2012. Association for Computational Linguistics.

[Callison-Burch, 2009] Chris Callison-Burch. Fast, cheap, and creative: evaluating translation quality using amazon's mechanical turk. In *EMNLP '09*, pages 286–295, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[Chen and Gey, 2003] Aitao Chen and Fredric C. Gey. Combining query translation and document translation in cross-language retrieval. In Carol Peters, Julio Gonzalo, Martin Braschler, and Michael Kluck, editors, *CLEF*, volume 3237 of *Lecture Notes in Computer Science*, pages 108–121. Springer, 2003.

[Chiang, 2005] David Chiang. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 263–270, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[Church and Hovy, 1993] Kenneth Ward Church and Eduard H. Hovy. Good applications for crummy machine translation. *Machine Translation*, 8(4):239–258, 1993.

[Condon *et al.*, 2010] Sherri L. Condon, Dan Parvaz, John S. Aberdeen, Christy Doran, Andrew Freeman, and Marwan Awad. Evaluation of machine translation errors in english and iraqi arabic. In *LREC*, 2010.

[Corston-Oliver *et al.*, 2001] Simon Corston-Oliver, Michael Gamon, and Chris Brockett. A Machine Learning Approach to the Automatic Evaluation of Machine Translation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 148–155, 2001.

[Darwish and Oard, 2003] Kareem Darwish and Douglas W. Oard. Probabilistic structured query methods. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 338–344, New York, NY, USA, 2003. ACM.

[Daumé and Marcu, 2006] Hal Daumé, III and Daniel Marcu. Bayesian query-focused summarization. In *ACL*, pages 305–312, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[de Gispert *et al.*, 2010] Adrià de Gispert, Gonzalo Iglesias, Graeme Blackwood, Eduardo R. Banga, and William Byrne. Hierarchical phrase-based translation with weighted finite-state transducers and shallow-n grammars. *Computational Linguistics*, 36(3):505—-533, 2010.

[Denkowski and Lavie, 2010] Michael Denkowski and Alon Lavie. Exploring normalization techniques for human judgments of machine translation adequacy collected using amazon mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 57–61, Los Angeles, June 2010. Association for Computational Linguistics.

[Denkowski and Lavie, 2011] Michael Denkowski and Alon Lavie. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*, 2011.

[Doyon *et al.*, 2008] Jennifer Doyon, Christine Doran, C. Donald Means, and Domenique Parr. Automated machine translation improvement through post-editing techniques: analyst and translator experiments. In *AMTA*, 2008.

[Elming, 2006] Jakob Elming. Transformation-based corrections of rule-based mt. In *EAMT-2006: 11th Annual Conference of the European Association for Machine Translation*, pages 219–226, 2006.

[Evans and McKeown, 2005] David Kirk Evans and Kathleen McKeown. Identifying similarities and differences across english and arabic news. In *International Conference on Intelligence Analysis*, 2005.

[Ferrández *et al.*, 2007] Sergio Ferrández, Antonio Toral, Óscar Ferrández, Antonio Ferrández, and Rafael Muñoz. Applying wikipedia's multilingual knowledge to cross-lingual question answering. In Zoubida Kedad, Nadira Lammari, Elisabeth Métais, Farid Meziane, and Yacine Rezgui, editors, *NLDB*, volume 4592 of *Lecture Notes in Computer Science*, pages 352–363. Springer, 2007.

[Finkel *et al.*, 2005] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[Fujii *et al.*, 2007] Atsushi Fujii, Makoto Iwayama, and Noriko Kando. Overview of the patent retrieval task at the ntcir-6 workshop. In Noriko Kando and David Kirk Evans, editors, *Proceedings of the Sixth NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering, and Cross-Lingual Information Access*, NTCIR-6, pages 359–365, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan, May 2007. National Institute of Informatics.

[Gamon *et al.*, 2005] Michael Gamon, Anthony Aue, and Martine Smets. Sentence-level MT Evaluation Without Reference Translations: Beyond Language Modeling. In *Proceedings of the European Association for Machine Translation (EAMT)*, 2005.

[Gao *et al.*, 2001] Jianfeng Gao, Jian-Yun Nie, Endong Xun, Jian Zhang, Ming Zhou, and Changning Huang. Improving query translation for cross-language information retrieval using statistical models. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on*

Research and development in information retrieval, pages 96–104, New York, NY, USA, 2001. ACM.

[Giménez and Màrquez, 2008] Jesús Giménez and Lluís Màrquez. A smorgasbord of features for automatic mt evaluation. In *StatMT '08: Proceedings of the Third Workshop on Statistical Machine Translation*, pages 195–198, Morristown, NJ, USA, 2008. Association for Computational Linguistics.

[Glenn *et al.*, 2011] Meghan Lammie Glenn, Lauren Friedman, Stephanie M. Strassel, Zhiyi Song, Gary Krug, Kazuaki Maeda, Haejoong Lee, and Christopher Caruso. *Handbook of Natural Language Processing and Machine Translation*, chapter Human Annotation, pages 14–64. Springer, 2011.

[Gonzalo *et al.*, 2005] Julio Gonzalo, Paul Clough, and Ro Vallin. Overview of the clef 2005 interactive track. In *In CLEF '05*, pages 251–262, 2005.

[Goto *et al.*, 2011] Isao Goto, Bin Lu, Ka Po Chow, Eiichiro Sumita, and Benjamin K. Tsou. Overview of the patent machine translation task at the ntcir-9 workshop. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access*, NTCIR-9, 2011.

[Guo *et al.*, 2009] Jiafeng Guo, Gu Xu, Xueqi Cheng, and Hang Li. Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 267–274, New York, NY, USA, 2009. ACM.

[Habash *et al.*, 2009] N. Habash, O. Rambow, and R. Roth. MADA+TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. *Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Cairo, Egypt*, pages 242–245, 2009.

[Habash, 2008] Nizar Habash. Four techniques for online handling of out-of-vocabulary words in Arabic-English statistical machine translation. In *Proceedings of ACL*, Columbus, OH, 2008.

[Habash, 2009] Nizar Habash. Remoov: A tool for online handling of out-of-vocabulary words in machine translation. In Khalid Choukri and Bente Maegaard, editors, *Proceedings of the Second International Conference on Arabic Language Resources and Tools*, Cairo, Egypt, April 2009. The MEDAR Consortium.

[Habash, 2010] Nizar Habash. *Introduction to Arabic Natural Language Processing.* Morgan & Claypool Publishers, 2010.

[Hakkani-Tür *et al.*, 2007] Dilek Hakkani-Tür, Heng Ji, and Ralph Grishman. Using Information Extraction to Improve Cross-lingual Document Retrieval. In Thierry Poibeau and Horacio Saggion, editors, *Proceedings of Multi-source, Multilingual Information Extraction and Summarization, RANLP*, 2007.

[He and Wu, 2011] Daqing He and Dan Wu. Enhancing query translation with relevance feedback in translingual information retrieval. *Information Processing & Management*, 47(1):1 – 17, 2011.

[Herbert *et al.*, 2011] Benjamin Herbert, Gyorgy Szarvas, and Iryna Gurevych. Combining query translation techniques to improve cross-language information retrieval. In Paul Clough, Colum Foley, Cathal Gurrin, Gareth Jones, Wessel Kraaij, Hyowon Lee, and Vanessa Mudoch, editors, *Advances in Information Retrieval*, volume 6611 of *Lecture Notes in Computer Science*, pages 712–715. Springer Berlin / Heidelberg, 2011.

[Hermjakob *et al.*, 2008] Ulf Hermjakob, Kevin Knight, and Hal Daumé III. Name translation in statistical machine translation - learning when to transliterate. In *Proceedings of ACL-08: HLT*, pages 389–397, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[Isabelle *et al.*, 2007] Pierre Isabelle, Cyril Goutte, and Michel Simard. Domain adaptation of mt systems through automatic post-editing. *MT Summit XI*, 2007.

[Ittycheriah and Roukos, 2007] Abraham Ittycheriah and Salim Roukos. Direct translation model 2. In Candace L. Sidner, Tanja Schultz, Matthew Stone, and ChengXiang Zhai, editors, *HLT-NAACL*, pages 57–64. The Association for Computational Linguistics, 2007.

[Järvelin and Kekäläinen, 2000] Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *SIGIR*, pages 41–48, 2000.

[Ji and Grishman, 2007] Heng Ji and Ralph Grishman. Collaborative entity extraction and translation. In *International Conference on Recent Advances in Natural Language Processing*, 2007.

[Jones *et al.*, 2007] Douglas Jones, Martha Herzog, Hussny Ibrahim, Arvind Jairam, Wade Shen, Edward Gibson, and Michael Emonts. Ilr-based mt comprehension test with multi-level questions. In *NAACL '07: Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers on XX*, pages 77–80, Morristown, NJ, USA, 2007. Association for Computational Linguistics.

[Kelly and Lin, 2007] Diane Kelly and Jimmy Lin. Overview of the trec 2006 ciqa task. *SIGIR Forum*, 41(1):107–116, 2007.

[Kirchhoff *et al.*, 2007] Katrin Kirchhoff, Owen Rambow, Nizar Habash, and Mona. Diab. Semi-automatic error analysis for large-scale statistical machine translation systems. In *Proceedings of the Machine Translation Summit IX (MT-Summit IX)*, 2007.

[Knight and Chander, 1994] Kevin Knight and Ishwar Chander. Automated postediting of documents. In *AAAI '94: Proceedings of the twelfth national conference on Artificial intelligence (vol. 1)*, pages 779–784, Menlo Park, CA, USA, 1994. American Association for Artificial Intelligence.

[Koehn *et al.*, 2007] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: open source toolkit for statistical machine translation. In *ACL '07: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180, Morristown, NJ, USA, 2007. Association for Computational Linguistics.

[Koehn, 2010] Philipp Koehn. *Statistical Machine Translation.* Cambridge University Press, New York, NY, USA, 1st edition, 2010.

[Kraaij, 2004] Wessel Kraaij. *Variations on Language Modeling for Information Retrieval.* PhD thesis, University of Twente, June 2004.

[Kulesza and Shieber, 2004] Alex Kulesza and Stuart M. Shieber. A Learning Approach to Improving Sentence-level MT Evaluation. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI)*, 2004.

[Kumaran and Allan, 2007] Giridhar Kumaran and James Allan. Information retrieval techniques for templated queries. In *Large Scale Semantic Access to Content (Text, Image, Video, and Sound)*, RIAO '07, pages 671–686, Paris, France, France, 2007. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

[Lavie and Agarwal, 2007] Alon Lavie and Abhaya Agarwal. Meteor: an automatic metric for mt evaluation with high levels of correlation with human judgments. In *StatMT '07: Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231, Morristown, NJ, USA, 2007. Association for Computational Linguistics.

[Lavrenko and Croft, 2001] Victor Lavrenko and W. Bruce Croft. Relevance-based language models. In W. Bruce Croft, David J. Harper, Donald H. Kraft, and Justin Zobel, editors, *SIGIR*, pages 120–127. ACM, 2001.

[Ma and McKeown, 2009] Wei-Yun Ma and Kathleen McKeown. Where's the verb?: correcting machine translation during question answering. In *ACL-IJCNLP '09: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 333–336, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[Magdy and Jones, 2011] Walid Magdy and Gareth J.F. Jones. An efficient method for using machine translation technologies in cross-language patent search. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 1925–1928, New York, NY, USA, 2011. ACM.

[Manning *et al.*, 2008] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schtze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[Mareček *et al.*, 2011] David Mareček, Rudolf Rosa, Petra Galuščáková, and Ondrej Bojar. Two-step translation with grammatical post-processing. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 426–432, Edinburgh, Scotland, July 2011. Association for Computational Linguistics.

[Mauser *et al.*, 2006] A. Mauser, R. Zens, E. Matusov, S. Hasan, and H. Ney. The rwth statistical machine translation system for the iwslt 2006 evaluation. In *International Workshop on Spoken Language Translation (IWSLT) 2006*, pages 103–110, Kyoto, Japan, November 2006. Best Paper Award.

[McCarley, 1999] J. Scott McCarley. Should we translate the documents or the queries in cross-language information retrieval? In *ACL*, 1999.

[Mehdad *et al.*, 2012] Yashar Mehdad, Matteo Negri, and Marcello Federico. Match without a referee: Evaluating mt adequacy without reference translations. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 171–180, Montréal, Canada, June 2012. Association for Computational Linguistics.

[Metzler and Croft, 2004] Donald Metzler and W. Bruce Croft. Combining the language model and inference network approaches to retrieval. *Inf. Process. Manage.*, 40(5):735–750, 2004.

[Miller and Beebe-Center, 1956] George A. Miller and J. G. Beebe-Center. Some psychological methods for evaluating the quality of translations. *Mechanical Translation*, 1956.

[Miller *et al.*, 1990] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. Introduction to wordnet: An on-line lexical database. *Journal of Lexicography*, 3(4):235–244, 1990.

[Nie and Jin, 2002] Jian-Yun Nie and Fuman Jin. A multilingual approach to multilingual information retrieval. In Carol Peters, Martin Braschler, Julio Gonzalo, and Michael Kluck, editors, *CLEF*, volume 2785 of *Lecture Notes in Computer Science*, pages 101–110. Springer, 2002.

[Nie, 2010] Jian-Yun Nie. *Cross-Language Information Retrieval.* Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2010.

[Nivre *et al.*, 2007] Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[Oard and Ertunc, 2002] Douglas W. Oard and Funda Ertunc. Translation-based indexing for cross-language retrieval. In *Proceedings of the 24th BCS-IRSG European Colloquium on IR Research*, pages 324–333, London, UK, 2002. Springer-Verlag.

[Oard and Gonzalo, 2002] Douglas W. Oard and Julio Gonzalo. The clef 2001 interactive track. In *Proceedings of CLEF 2001, Springer-Verlag LNCS Series*, pages 203–214, 2002.

[Oard and Gonzalo, 2003] Douglas W. Oard and Julio Gonzalo. The clef 2003 interactive track. In *Proceedings of the Fourth Cross-Language Evaluation Forum. 2003.* Springer, 2003.

[Oard *et al.*, 2008] Douglas W. Oard, Daqing He, and Jianqiang Wang. User-assisted query translation for interactive cross-language information retrieval. *Inf. Process. Manage.*, 44(1):181–211, 2008.

[Oard, 1998] Douglas W. Oard. A comparative study of query and document translation for cross-language information retrieval. In David Farwell, Laurie Gerber, and Eduard H. Hovy, editors, *AMTA*, volume 1529 of *Lecture Notes in Computer Science*, pages 472–483. Springer, 1998.

[Och and Ney, 2000] Franz Josef Och and Hermann Ney. A comparison of alignment models for statistical machine translation. In *Proceedings of the 18th conference on Computational linguistics - Volume 2*, COLING '00, pages 1086–1090, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[Och and Ney, 2003] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[Och, 2012] Franz Och. Breaking down the language barrier—six years in. http://googleblog.blogspot.com/2012/04/breaking-down-language-barriersix-years.html, 2012.

[Owczarzak *et al.*, 2007] Karolina Owczarzak, Josef van Genabith, and Andy Way. Labelled dependencies in machine translation evaluation. In *ACL 2007: proceedings of the Second Workshop on Statistical Machine Translation*, pages 195–198, Morristown, NJ, USA, 2007. Association for Computational Linguistics.

[Papineni *et al.*, 2002a] Kishore Papineni, Salim Roukos, Todd Ward, John Henderson, and Florence Reeder. Corpus-based comprehensive and diagnostic mt evaluation: initial arabic, chinese, french, and spanish results. In *HLT*, pages 132–137, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[Papineni *et al.*, 2002b] Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.

[Parton *et al.*, 2009] Kristen Parton, Kathleen R. McKeown, Bob Coyne, Mona T. Diab, Ralph Grishman, Dilek Hakkani-Tür, Mary Harper, Heng Ji, Wei Yun Ma, Adam Meyers, Sara Stolbach, Ang Sun, Gokhan Tur, Wei Xu, and Sibel Yaman. Who, what, when, where, why?: comparing multiple approaches to the cross-lingual 5w task. In *ACL-IJCNLP '09*, pages 423–431, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[Pirkola, 1998] Ari Pirkola. The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval. In *SIGIR*, pages 55–63. ACM, 1998.

[Popović and Ney, 2007] Maja Popović and Hermann Ney. Word error rates: Decomposition over POS classes and applications for error analysis. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 48–55, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[Quirk, 2004] Christopher Quirk. Training a Sentence-level Machine Translation Confidence Measure. In *Proceedings of LREC*, 2004.

[Raybaud *et al.*, 2011] Sylvain Raybaud, David Langlois, and Kamel Smaïli. "this sentence is wrong." detecting errors in machine-translated sentences. *Machine Translation*, 25:1–34, 2011. 10.1007/s10590-011-9094-9.

[Resnik and Smith, 2003] Philip Resnik and Noah A. Smith. The web as a parallel corpus. *Comput. Linguist.*, 29(3):349–380, September 2003.

[Rosa *et al.*, 2012] Rudolf Rosa, David Mareček, and Ondřej Dušek. Depfix: A system for automatic correction of czech mt outputs. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 362–368, Montréal, Canada, June 2012. Association for Computational Linguistics.

[Sadat and Habash, 2006] Fatiha Sadat and Nizar Habash. Combination of arabic preprocessing schemes for statistical machine translation. In *Proceedings of the Conference of the Association for Computational Linguistics*, Sydney, Australia, 2006.

[Simard *et al.*, 2007] Michel Simard, Cyril Goutte, and Pierre Isabelle. Statistical phrase-based post-editing. In *HLT-NAACL*, pages 508–515, 2007.

[Snover *et al.*, 2006] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *In Proceedings of Association for Machine Translation in the Americas*, pages 223–231, 2006.

[Snover *et al.*, 2009] Matthew Snover, Nitin Madnani, Bonnie J. Dorr, and Richard Schwartz. Fluency, adequacy, or hter?: exploring different human judgments with a tunable mt metric. In *StatMT '09: Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 259–268, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[Snow *et al.*, 2008] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *EMNLP '08*, pages 254–263, Morristown, NJ, USA, 2008. Association for Computational Linguistics.

[Soricut and Echihabi, 2010] Radu Soricut and Abdessamad Echihabi. Trustrank: Inducing trust in automatic translations via ranking. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 612–621, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[Specia *et al.*, 2011] Lucia Specia, Najeh Hajlaoui, Catalina Hallett, and Wilker Aziz. Predicting machine translation adequacy. In *MT Summit XIII*, 2011.

[Strohman *et al.*, 2005] Trevor Strohman, Donald Metzler, Howard Turtle, and W. Bruce Croft. Indri: A language-model based search engine for complex queries (extended version). IR 407, University of Massachusetts, Amherst, University of Massachusetts, 2005.

[Stymne and Ahrenberg, 2010] Sara Stymne and Lars Ahrenberg. Using a grammar checker for evaluation and postprocessing of statistical machine translation. In *LREC*, 2010.

[Suzuki, 2011] Hirokazu Suzuki. Automatic post-editing based on smt and its selective application by sentence-level automatic quality evaluation. *MT Summit XIII*, 2011.

[Systran, 2012] Systran. Systran hybrid technology. http://www.systransoft.com/systran/corporate-profile/translation-technology/systran-hybrid-technology, 2012.

[Toutanova *et al.*, 2003] Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 173–180, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[Ueffing and Ney, 2005] Nicola Ueffing and Hermann Ney. Word-level confidence estimation for machine translation using phrase-based translation models. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 763–770, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[Ueffing *et al.*, 2008] Nicola Ueffing, Jens Stephan, Evgeny Matusov, Loïc Dugast, George Foster, Roland Kuhn, Jean Senellart, and Jin Yang. Tighter integration of rule-based and statistical mt in serial system combination. In *COLING '08: Proceedings of the 22nd International Conference on Computational Linguistics*, pages 913–919, Morristown, NJ, USA, 2008. Association for Computational Linguistics.

[Uszkoreit *et al.*, 2010] Jakob Uszkoreit, Jay Ponte, Ashok Popat, and Moshe Dubiner. Large scale parallel document mining for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1101–1109, Beijing, China, August 2010. Coling 2010 Organizing Committee.

[Vilar *et al.*, 2006] David Vilar, Jia Xu, Luis Fernando D'Haro, and Hermann Ney. Error analysis of machine translation output. In *International Conference on Language Resources and Evaluation*, pages 697–702, Genoa, Italy, May 2006.

[Vogel *et al.*, 2003] Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venugopal, Bing Zhao, and Alex Waibel. The cmu statistical machine translation system. In *Proceedings of MT Summit IX*, 2003.

[Wang and Oard, 2001] Jianqiang Wang and Douglas W. Oard. iclef 2001 at maryland: Comparing term-for-term gloss and mt. In Carol Peters, Martin Braschler, Julio Gonzalo, and Michael Kluck, editors, *CLEF*, volume 2406 of *Lecture Notes in Computer Science*, pages 336–354. Springer, 2001.

[Wang and Oard, 2006] Jianqiang Wang and Douglas W. Oard. Combining bidirectional translation and synonymy for cross-language information retrieval. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 202–209, New York, NY, USA, 2006. ACM.

[Wu *et al.*, 2008] Dan Wu, Daqing He, Heng Ji, and Ralph Grishman. A study of using an out-of-box commercial mt system for query translation in clir. In *iNEWS '08: Proceeding of the 2nd ACM workshop on Improving non english web searching*, pages 71–76, New York, NY, USA, 2008. ACM.

[Xu and Weischedel, 2000] Jinxi Xu and Ralph Weischedel. Trec-9 cross-lingual retrieval at bbn. In *In TREC-9*, pages 106–116, 2000.

[Zeman *et al.*, 2011] Daniel Zeman, Mark Fishel, Jan Berka, and Ondrej Bojar. Addicter: What is wrong with my translations? *Prague Bull. Math. Linguistics*, 96:79–88, 2011.

[Zhang *et al.*, 2009] Yuqi Zhang, Evgeny Matusov, and Hermann Ney. Are unaligned words important for machine translation ? In *Conference of the European Association for Machine Translation*, pages 226–233, Barcelona, March 2009.