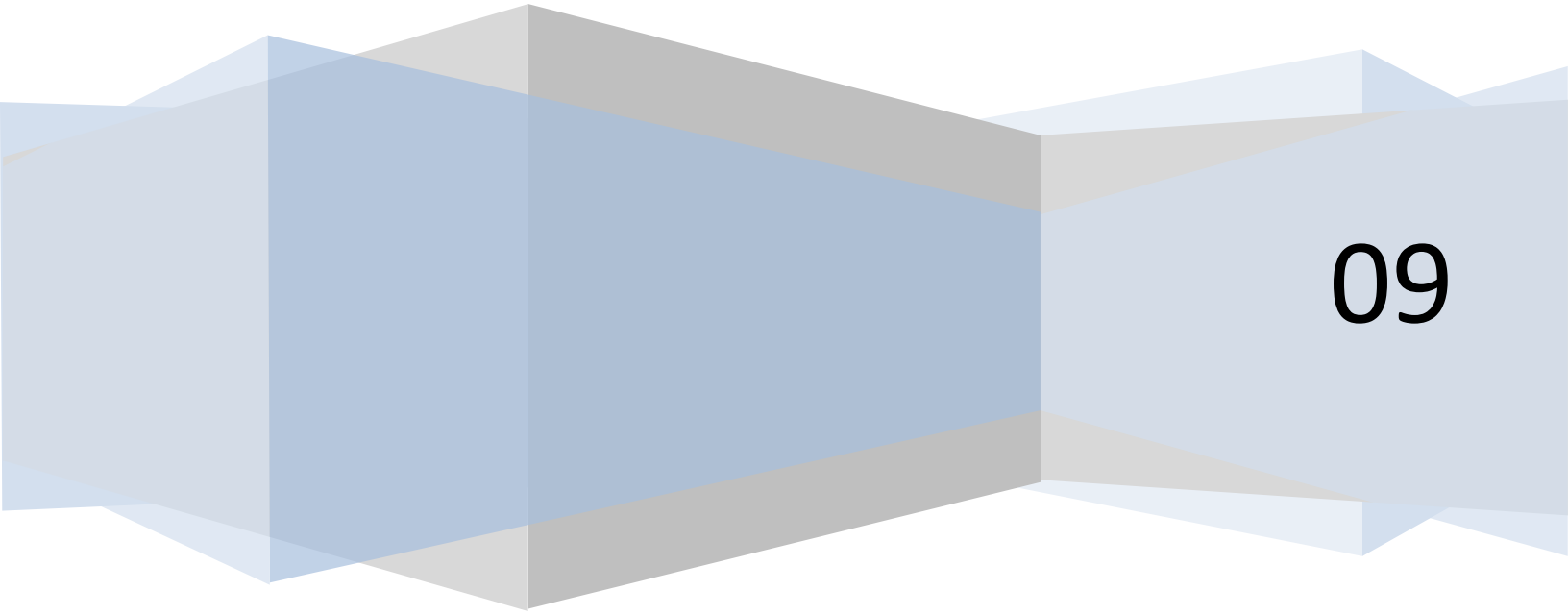
University of Washington Department of Computer Science

Practical Translation for All Languages

The Design, Implementation and Evaluation of Panlingual Translator

By Christopher Lim



09

Abstract

We describe the design, implementation and evaluation of a real Lemmatic Machine Translation (LMT) system called the Panlingual Translator, which enables us to investigate the ability of non-expert users to practically communicate using the LMT approach and the particular conditions that enable LMT to work. We discover that naïve users can adequately convey their meaning using LMT when a minimal set of key terminology is correctly translated to communicate intrinsic genre, which then enables shared domain knowledge to correct any errors in meaning transmission.

Introduction

Since the Tower of Babel, the need for language translation has exponentially increased with most knowledge and communication happening in only a handful of majority languages. Furthermore, the advent of the web has accelerated the pace of content creation and this content is currently valuable only to people who speak the language it is produced in.

Machine Translation has promised to meet human translation needs for decades, but has suffered from a lack of sufficient linguistic data to learn from. People expecting fluent translations have been frequently disappointed by the quality of machine translation, relegating its usage to scenarios like providing "gist translations" of documents on the web. Even the progress that has been made to date has focused on about the top 55 languages in the world, leaving the needs of speakers of the over 6,000 remaining living languages unmet[1] (1).

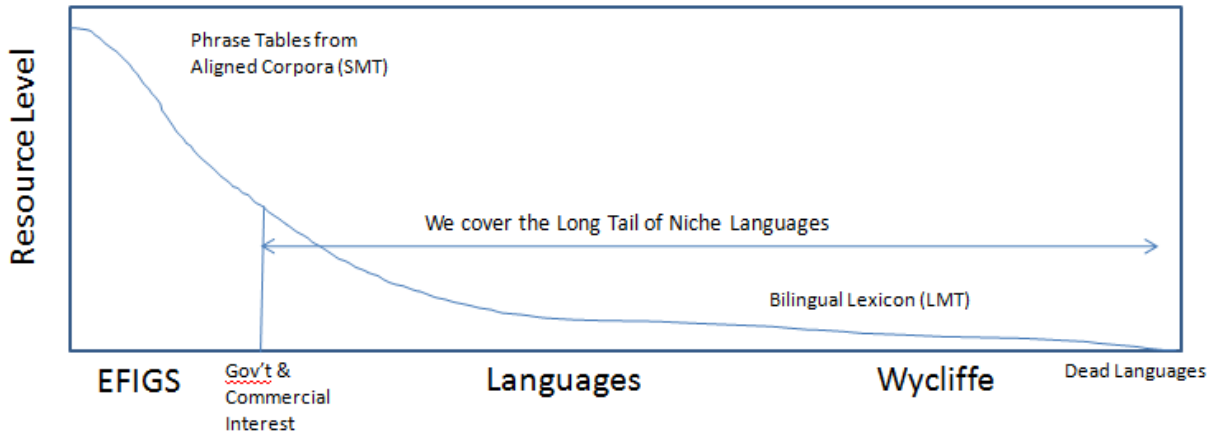


Figure 1 Qualitative Graph showing the range of languages LMT is intended to serve and the linguistic resource level.

In order to support translation for low-density languages, some have tried accelerating the development of machine-readable linguistic data while others have tried reducing the amount of manual effort and

[1] For example, the state of the art machine translation system Google Translate only supports about 55 of the over 6,000 living languages extant.

data required (2). We propose Lemmatic Machine Translation (LMT)[2], which is lexical translation of sequences of dictionary words[3], as an alternative approach that scales to support translation between all languages in the world. By reducing the data requirements to basic bilingual dictionaries, LMT trades fluency to gain adequacy for a higher coverage of languages and also leverages user interactivity for word sense disambiguation.

Previous work by Soderland et al. (3) has demonstrated the adequacy of LMT when used by experts. An experienced user of LMT encodes messages in short simple sentences using dictionary-form words. The message is translated word for word into another language and when all words are translated with high probability, about 93% of sentences are properly understood.

Following up on their work, we investigate non-expert users' ability to adequately convey meaning using LMT. Is this approach practically useful for communication or is it a mere novelty? To answer the question, we built the Panlingual Translator system[4], a website where people can write a message, get a lexical translation and interactively improve it without knowing anything about their target language. Our informal evaluations indicate that LMT is a practical solution that enables naïve users to adequately translate between all languages in the world.

After observing the practical adequacy of LMT, we also explored the reasons why word-for-word translation of dictionary words can adequately convey meaning. The most surprising result was that the lemmaticality[5] of a message did not strongly correlate with its translation adequacy, implying that simplified, broken speech is not what mainly helps –word-for-word translations of fluent messages were also adequately translated. Based on anecdotal evidence, we conjecture that LMT works when enough terminology is unambiguously translated to identify the message's intrinsic genre, at which point a recipient's domain knowledge overcomes any translation problems. This insight leads us to recommend several best practices that increase the likelihood for adequate translation when using LMT.

The rest of this thesis is structured as follows. We begin by outlining a history of related work followed by an explanation of the LMT paradigm. Then we describe the Panlingual Translator's design and implementation history, showing how it embodies knowledge gained from previous work and describing our solutions to the new problems that emerged while building a practical system for naïve users. Finally, we conclude with the results from an informal user study that demonstrate the utility and limitations of Panlingual Translator while offering suggestions for future work.

Related Work

The field of machine translation is traditionally categorized into three paradigms: Rule-Based (RBMT), Statistical (SMT), Example-Based (EBMT), and hybrids of these three (4). All of these approaches aim for

[2] For colloquial clarity, we could refer to LMT as "Broken Speech Machine Translation" or BSMT
[3] We mean words and phrases in general.
[4] Currently available at http://lepton.cs.washington.edu:8080/Lemuel/
[5] Lemmaticality is a subjective measure of a sentence's simplicity, shortness and use of simple, translatable vocabulary

fluent translations through different means. RBMT requires human encoding of many complex linguistic rules, EBMT adapts example translations from bilingual corpora into novel ones, SMT generalizes from bilingual corpora to compose new translations and hybrids combine these approaches to improve output quality while reducing manual labor and data requirements. Unfortunately, the resources required to gain fluent automatic translations remains out of reach for most of the world's over 6,000 languages (2) and even with abundant resources, people are often disappointed by the output quality.

In contrast, Lemmatic Machine Translation serves the long tail of languages by reducing data requirements to basic bilingual dictionaries to get *adequate word-for-word translations*, which successfully convey meaning without attempting to be fluent. Dictionary based word-for-word translation has been attempted since the 1950s (5) and was recently used as a component of a hybrid system by Carbonell et al. (6). Unlike Carbonell's approach, which requires a full-form bilingual dictionary, our approach uses basic translation dictionaries of surface strings.

LMT is also a form of human-aided machine translation (7) because it leverages user interactivity for word sense disambiguation and lemmatic encoding of messages. Users compose messages with words from a large domain-independent vocabulary, which are then translated for human decoding. This distinguishes LMT from traditional controlled language scenarios that parse a restricted vocabulary into an interlingua (8).

Within the LMT paradigm, this study follows a line of work started in 2005 with the inception of the Turing Center whose mission is to advance universal communication between humans and artificial agents. Seminal work by Etzioni et al. (9) renewed interest in lexical translation, which peaked in the 1990s (10), by demonstrating its practical utility for image search. They argued that the proliferation of Machine Readable Dictionaries (MRDs), the rapid growth of multilingual Wiktionaries, and the increasingly international web audience presented many opportunities for novel applications of lexical translation systems. They also described the TransGraph system, which combined independently authored MRDs and Wiktionaries into a graph on which they performed online probabilistic inference to select translations.

This work was accompanied by research into building a sense-distinguished multilingual lexicon (11) and what ultimately emerged from the two approaches was a sense-distinguished multilingual dictionary called the PanDictionary (12). The PanDictionary[6] uses probabilistic inference to find translations that are not in any of its source dictionaries and we selected it for use in Panlingual Translator because it enables our system to translate between languages where no bilingual lexicons or parallel texts exist.

Recent work by Soderland et al. (3) has evaluated the translation adequacy of LMT as embodied in the Panlingual Translator discovering that *expert users* of the system achieve translation adequacy in 93% of sentences where only high probability translations are used. Their experiment was conducted for nine language pairs, including three not supported by Google Translate, and four different message genres. Continuing this work, we attempt to build a practical system that lets *naïve user*s achieve similar

[6] The PanDictionary has 10 million words in more than 1,000 languages and 81,000 word senses. At an estimated precision of 0.90 it has over 1.6 million translations in over 700 languages.

translation adequacy and investigate the reasons why LMT is even possible in order to discover the best practices for using it.

An initial study was conducted on the viability of lemmatic communication by Everitt et al. (13), which tested the effects of lemmatization, word ordering and translation on successful meaning transfer. They discovered that while non-grammatical simplified sequences of dictionary form words did not disrupt meaning transfer, randomized word re-ordering significantly did, disrupting about 60% of test sentences when combined with translation. That study presented several recommendations for developing a practical lemmatic translation user interface like creating a free input system with source language feedback, which we implemented in the Panlingual Translator.

Christensen et al. (14) studied forms of feedback for user-assisted word sense disambiguation discovering that definitions were the most accurate method closely followed by synonyms and lastly images, which were often ambiguous when shown alone. They advocated showing all available information for a word, recommending images labeled with synonyms because it provided an enjoyable and accurate disambiguation user experience. We chose to offer two backtranslations as feedback for word sense disambiguation in the Panlingual Translator system because it removed the problem of finding images to match each sense and could easily provide feedback in any language unlike definitions which need to be translated.

With respect to word sense disambiguation, Soderland et al. also describes an automatic corpus-based approach that performed better than the baseline translation, but did worse than the interactively disambiguated translations. Since we built an interactive system, this technique was not incorporated in the Panlingual Translator. Christensen et al. (15) later explored automated methods for automatic implicit word sense disambiguation through the selection of optimal translations that return good image results, which may provide better disambiguations if incorporated into our system.

Panlingual Translator embodies the best practices learned from these previous studies into a practical system that supports lemmatic encoding, implicit word sense disambiguation, and lexical translation. We also evaluated our system by conducting an informal user study that tested naïve users' ability to learn, use and adequately translate with it. The following section describes the process of lemmatic translation and how we decided to support it in the Panlingual Translator.

Lemmatic Translation

The Intuition

Following the hermeneutical theory of Hirsch (16), we view communication (reading in particular)[7] as a process of re-cognition, where an author serializes an intended meaning into a sequence of words,

[7] We use the terms "speaker" and "author" interchangeably in this section, but Panlingual Translator is a system for translating text only at this time. We also view asynchronous translated messages as part of a "conversation" even though our system does not currently support real-time conversation translations.

which a recipient then interprets to reconstruct the intended meaning while iteratively refining the interpretation through questioning and conversation[8].

The complexity of language, which makes translation so difficult, enables it to convey a rich set of precise meanings in a compact form. The simplified form used in lemmatic translation, which produces ungrammatical utterances composed of translated dictionary words, is unable to actualize the same meaning possibilities with the same precision and the same density as full natural language.

For example, a hotel reservation request could be written in English with a single sentence: "I want to book a two person room for two nights starting on December 23rd." The English syntax imposes a structure on this sequence of words that give it a clear meaning. A lemmatic encoding of the same message on the other hand, might require several shorter, simpler sentences: "I want room / two people / two night / start December 23rd". While it may require more interpretive work on the part of a message recipient, this lemmatic example and many other instances can practically convey meaning.

In general, we observe the phenomenon of travelers in a foreign country with no grammatical knowledge being able to effectually communicate by stringing together short sequences of translated dictionary words. Even intralingually, we see children able to communicate with adults using a small lexicon and minimal grammar. This leads us to believe that the loss of meaning possibilities, precision and density in lemmatic communication is acceptable and can be overcome through other means such as lexicalizing meanings encoded in word inflection, using gestures and facial expressions or grounding the communication in pictures, objects, etc. Thus, because of its adequacy and low resource requirements, LMT may be the most practical way to support translation between the over 6,000 low-density languages in the world[9].

The Process

The actual process of lemmatic translation depends on the way we frame the communicative interaction. When we created the first design of the Panlingual Translator, we envisioned the interaction as in Model 1 of Figure 2. Two people converse with each other while our system passively provides translations of the authors' message. The computer is just one aid out of many cues like immediate bodily and facial expression feedback that participants use to reconstruct each others' intended meaning. Unfortunately, this view led to an unintuitive user interface, which provided awkward feedback and tools because LMT requires active human participation in lemmatic encoding and word sense disambiguation (i.e. it is human-assisted machine translation) while the interaction model relegated the computer to a passive role as a communicative aid (computer aided translation).

---

[8] The interpretative process begins with a hypothesized "extrinsic genre", or set of meanings, which provide a set of "meaning expectations" that determine how we interpret and disambiguate an incoming word sequence. This set of meaning expectations is adjusted probabilistically as we gain new evidence from the text until we arrive at our best hypothesis for the text's "intrinsic genre" or the particular meanings the author intended for that particular text. Thus interpretation is like searching through a space of "sense/meaning sets" until we arrive at one that is consistent with the "intrinsic genre" of a text.

[9] LMT is particularly suited for human communication when people are physically present, sharing a rich context of shared meanings with many nonlinguistic means (i.e. gestures, acting, pointing) to make up for linguistic loss.

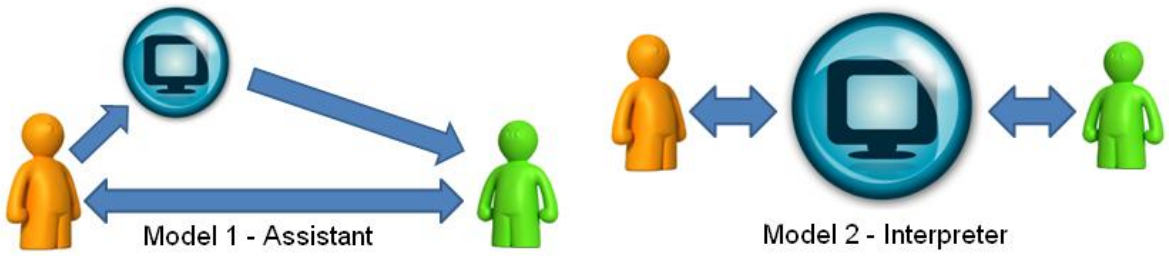


Figure 2: In our original Model 1 view, the sender primarily interacts with the recipient, the computer providing a rough translation without human cooperation. In Model 2, the sender interacts mainly with the computer, confident that if the computer "understands" the sender's meaning, the recipient will get an adequate translation.

In contrast, we now view the computer as a conversation partner (see Model 2 above). A user communicates with the computer until it "understands" their intended meaning and Panlingual Translator then conveys this meaning in another language to the message recipient. This interaction model results in a logical user interface: users are given feedback to explore the Panlingual Translator's lexicon and understanding, the way they might ask questions to explore another human being's vocabulary and comprehension. Since people experiment with paraphrases and synonyms when their partners cannot understand their original messages, we provide suggested words and easy experimentation features so users can quickly iterate through all possible ways of communicating their ideas until they find a satisfactory encoding. Lastly, we let users correct the computer's interpretation of their message by providing them with backtranslation feedback, which users can fix by selecting a word with better backtranslations.

This process can be summarized as follows:

1. Compose your message
2. Discover which words were not understood
3. Decide what to do about these words (Ignore, Define, Try alternate words, Rephrase)
4. Discover which words were misunderstood
5. Decide what to do about these words (select correct meaning or follow options in step 3)
6. Repeat 2-5 until no more important meanings are not understood or misunderstood
7. Take the translation output and use it

Words that are not understood are missing from Panlingual Translators lexicon. Words that are misunderstood are incorrectly disambiguated. An example of the former is "microwave", for which we had no translation into Indonesian. Users in our evaluation always chose to define the purpose of a microwave and leave the word untranslated thereby "adding" it to the message recipient's lexicon. An example of the latter is "right" a term for which Panlingual Translator only offered translations through the sense of justice and correctness without any available translations into Indonesian through the sense related to direction. When users encountered this problem in our evaluation, some simply dropped the word while others tried synonyms like "east".

We now describe the design evolution of Panlingual Translator including the key issues we faced when trying to implement the LMT system and our solutions to those problems. The overarching requirement

was that each of the following iterations had to support the Lemmatic Translation process described above. What varied between the versions was the way we support the process, the scope of the system, and the unique problems that arose with each new design.

The Panlingual Translator

What we set out to build: PanMail

Building on momentum from the PanImages system described in (9), we began work on a Panlingual e-mail website in order to demonstrate another useful application of lexical translation. We wanted to enable people to compose lemmatic messages, manually disambiguate their messages, get the lexical translation and send the translated e-mail. Recipients of the message would be directed to a public interface where they could click on words in the e-mail to see the sender's disambiguation or alternate translations to help them understand and respond. A screenshot of the website is shown below.

Figure 3: A screenshot of the inbox view in "PanMail - E-mail in any language!"

This prototype supported internal messaging between registered users of the system so we primarily used it as a convenient way to explore the PanDictionary and compose test messages for our early evaluations. The lemmatic encoding interface shown in Figure 4 is a compromise design that attempts to accomplish several things. First, it gives the encoder feedback on words that can be translated by showing a dropdown of possible completions of the current word they are typing. Second, it avoids frustrating users by letting them type sentences freely. Last, it enforces lemmatic encoding by breaking up what would normally be a freeform textbox into individual single-line textboxes. The tension between freeform input and enforced lemmaticality is a consistent theme throughout all of our designs. We believed that the lemmaticality of source messages directly affected the adequacy of LMT output and therefore sought to promote it as much as possible.

The previous approach to enforcing lemmatic encoding was the design described in Everitt et al.'s paper and shown in Figure 5. Users were required to input one word per textbox and new textboxes were dynamically created as needed. Each textbox provided autocompletions and required users to select words translatable through the PanDictionary before moving on. Because of deep user frustration over being forced to deal with translation problems up front, Everitt et al. recommended against restricting user input leading us to develop the compromise design.

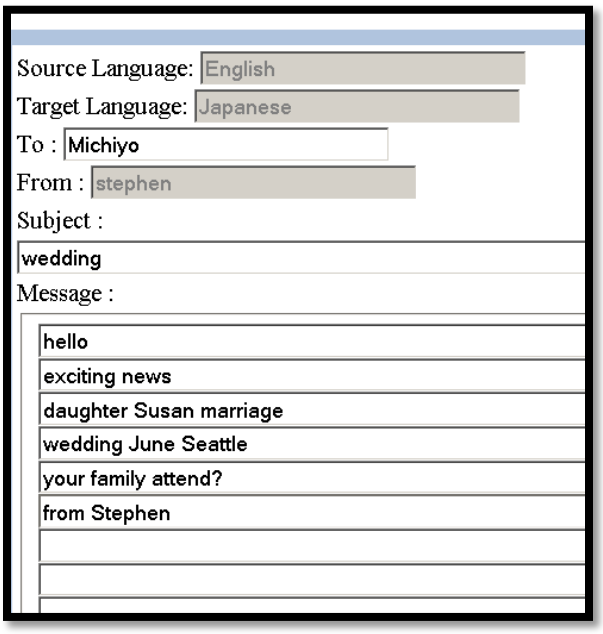


Figure 4: To enforce lemmatic encoding without frustrating users our original interface required line-by-line composition.

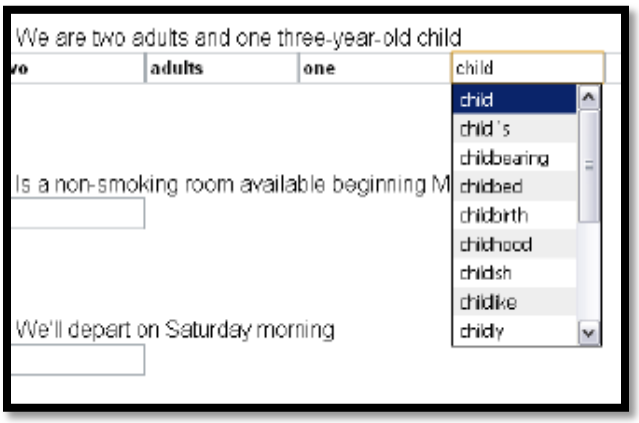


Figure 5: The design in Everitt et al. enforced lemmatic encoding by requiring users to compose word-by-word with one translatable word per input box. Users could not go to the next word until inputting a translatable word in the current box.

After the initial composition phase, Panlingual Mail tokenized the message and presented the user with an interface similar to the one in Figure 6, which let them click on tokens to see a list of senses (with English definitions and panlingual[10] backtranslations) and the highest probability translation of the sense. Words that had no translations were highlighted in red, words with low probability translations were highlighted in yellow and manually disambiguated words were highlighted in green. When the user was satisfied with their message (because they had added/deleted tokens and disambiguated mistranslated ones) they clicked send and it appeared in the recipient's inbox.

[10] "Panlingual" means "all languages" or "any language". In this case, we mean that the backtranslations can be in any language (whatever source language the user composes in) while the definitions are only available in English.

Figure 6: A screenshot of the translation viewing interface. The preview interface when composing a message is similar.

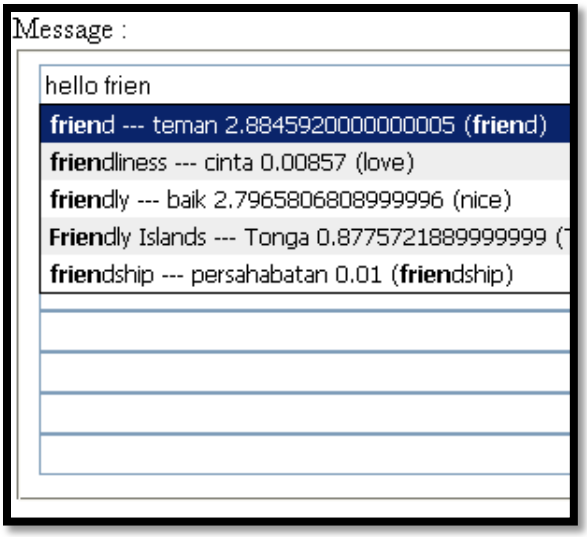


Figure 7: The autocomplete dropdown feedback we gave was "word --- translation probability (backtranslation)"

The recipient could then view the original message inline or click on words to see alternate translations (and the probability of each alternative) to help them reconstruct the intended meaning. The tokens also retained their highlighting so recipients could see which tokens senders disambiguated and were very sure about and which ones were algorithmic selections.

The first algorithm we tried for selecting the best translation was as follows: for a given source word w_s in language l_s, we select the sense that it belongs to with highest probability $s^* = argmax_s p(w_s \epsilon s)$. We then select the highest probability translation of the sense in the target language l_t obtaining $BestTrans(w_s, l_t) = argmax_{w_t} p(w_t \epsilon s^*)$. This simple algorithm made a lot of mistakes because the PanDictionary had many instances of a word equally belonging to several senses, in which case we arbitrarily selected one.

From a usability perspective, we discovered the following issues with this design:

- Sense disambiguation through definitions cannot easily be provided in all languages
- Iteratively improving a message through the preview panel was difficult and users preferred being able to go back to the initial composition screen to make changes instead of editing individual tokens.
- The autocompletions were plagued with performance issues because it started looking up words in our massive dictionary after a single character was entered and would then return an enormous result set. This bogged down the interface so that waiting for autocompletions became a hindrance rather than a help. Everitt et al. also discovered that people preferred to solve translation problems after their message is composed rather than during composition.
- Translation took too long to be useful (people could not interactively adapt their message). Wait times of 4 minutes were common because of the way the system was designed. It would run SQL Server stored procedures to do the translations over the complete dictionary on the fly.

Even after adding good indices to the database, this could take on the order of 5 minutes to translate a message like in Figure 4.

On the other hand, this system also had several advantages like persistent storage of messages for future analysis, full disclosure of all the available information in the PanDictionary (e.g. definitions, parts of speech), and an interface for helping recipients interpret messages. Implementing this design also gave us a platform to test translation selection algorithms and the opportunity to familiarize ourselves with JSP, JQuery, and the details of the PanDictionary. Overall, it focused on functionality over usability.

During this time, we also began evaluating the translation adequacy of LMT by translating four test messages from English into various languages and asking bilingual workers on Amazon's Mechanical Turk (AMT) to translate our messages back into English. AMT is an online marketplace where requesters post tasks with a micropayment price for the completion of the job. Workers accept these tasks and are paid upon satisfactory completion.

While having easy access to this large diverse workforce was beneficial, we unfortunately found it very difficult to use AMT because of numerous illegitimate results we had to manually screen out. We discovered that there were few minority language speakers using the platform (i.e. we quickly ran out of fresh bilingual informants), paying a premium attracted a lot of attention ($2.50/job), few people followed the precise directions we posted, many tried to cheat by fulfilling our jobs several times even after we asked them not to or by using automatic translation systems, and the website itself had poor Unicode support and a buggy interface for sifting through results.

Our Second Attempt: Panlingual Translator Alpha

After familiarizing ourselves with available technology and discovering the user interaction issues from our first design, we decided to throw away the PanMail prototype and rebuild from scratch[11]. Having fulfilled the functional requirements of LMT, we now focused on enabling practical translation for non-expert users in a production system. Therefore, we reduced the scope and complexity of the project by dropping the e-mail aspect. The new product, Panlingual Translator, would still enable people to compose a message and interactively adapt it until they were satisfied with its perceived translation adequacy. However, it would no longer offer the trappings of an e-mail system like sending and reading messages or personal inboxes.

Our design goals for this new website were a painless, robust, real-time system that lets people effectively translate into languages unsupported by Google Translate. This meant that recipients of a message should be minimally burdened and encoders should have a delightful experience crafting their message and translation in a very responsive system.

Towards these ends, we focused first on providing the right kind of feedback in ways that raised user awareness of problems without being forced to deal with them up front. Second, we wanted to create a

[11] Following Fred Brooke's recommendation to "throw one away" and the philosophy of rapid prototyping+iterative development

logical way for users to deal with the translation problems they might encounter, and third, we wanted to significantly speed up translation time to enable true interactivity.

According to Christensen et al. (14), images of a word with synonym labels gave users enjoyable and useful feedback that could easily work in any language. Unfortunately, the image retrieval problem is difficult in its own right, so we opted to only provide backtranslation feedback.

Our algorithm for translation and backtranslation selection was described in Soderland et al. and is repeated here:

> *"The AllSenses translation method we use in Panlingual Translator combines evidence from multiple PanDictionary senses. For each possible translation w_t of w_s into language l_t, sum the probability that both w_s and w_t share sense s over all senses that contain both words. Return the w_t that has the maximum total probability."* (3)

$$AllSenses(w_s, l_t) = \underset{w_t}{\operatorname{argmax}} \left(\sum_{s \in senses} pr(w_s \in s) * pr(w_t \in s) \right)$$

A backtranslation simply imitates this process with reversed arguments where w_s becomes w_t, the translation of a word in language l_t, and l_t becomes l_s or the source language. Translating a word back into the source language through appropriate senses into new words gives us synonym-like feedback. We selected the top two backtranslations as feedback to help users determine the implicit sense of a translation, since a single word remains very ambiguous. For simplicity, we did not display definitions, parts of speech, or probabilities even when they were available.

Three different mock-up UI designs are shown in Figure 8 to Figure 10.

Figure 8: A design like the PanMail encoding interface

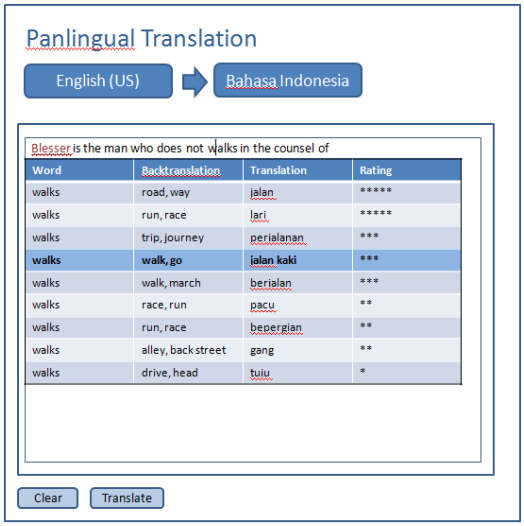


Figure 9: A freeform input design with autocompletions

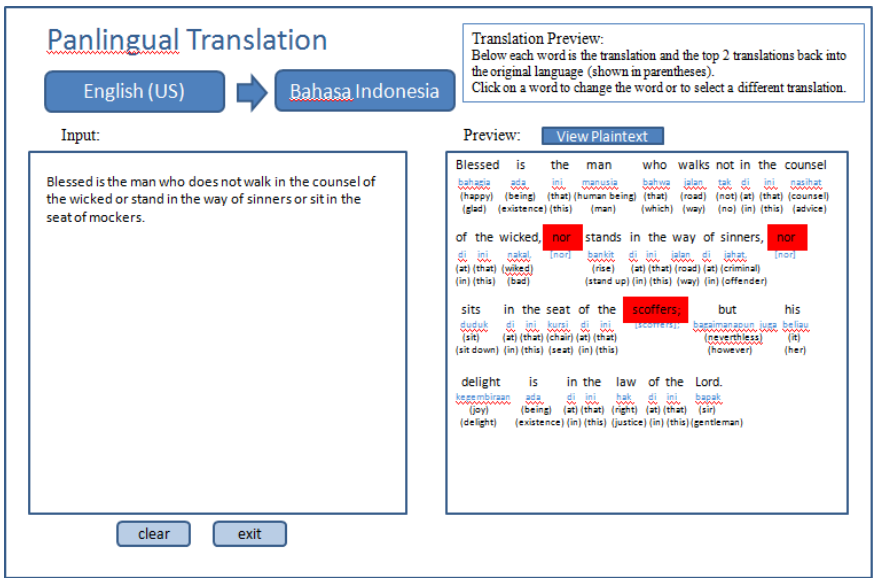


Figure 10: A design offering freeform input without autocompletion instead showing simultaneous feedback

The design in Figure 8 had a lot of the same problems as the original PanMail prototype and while the design in Figure 9 seemed ideal because of its freeform input, it hid useful feedback behind autocomplete dropdowns, which we also found to be difficult to robustly implement. The design in Figure 10 tried to combine the benefits of freeform input with the benefits of fully visible feedback. Unfortunately it made users click a button to get a copy-and-pastable translation output and seemed to show an overwhelming amount of information.

After some team deliberation, we decided on another compromise design that tried to balance the tension between overwhelming and underwhelming the user with feedback and forcing the user to lemmatically encode their message versus supporting freeform input. We resolved the tension by showing a lot of information for a single sentence and forcing the user to compose sentence by sentence, but giving them free reign within each individual sentence. The three mock-ups we designed for this interface are shown in Figure 11.

Instead of forcing users to correct translation problems up front, we let them write a complete sentence first, then translate it, examine the feedback and fix it. Unlike the PanMail prototype, users edit their message in the same place that they compose their message—the sentence input box. To raise user awareness of translation problems, we show in a preview box, their source sentence, the translation, and two backtranslations. When users are done, they click “New Sentence”, which adds their current sentence and its translation to the output boxes while clearing out the sentence input box. This design can be viewed like an Instant Messenger conversation window where people type and submit short messages, which can be seen in the history, but no longer edited.

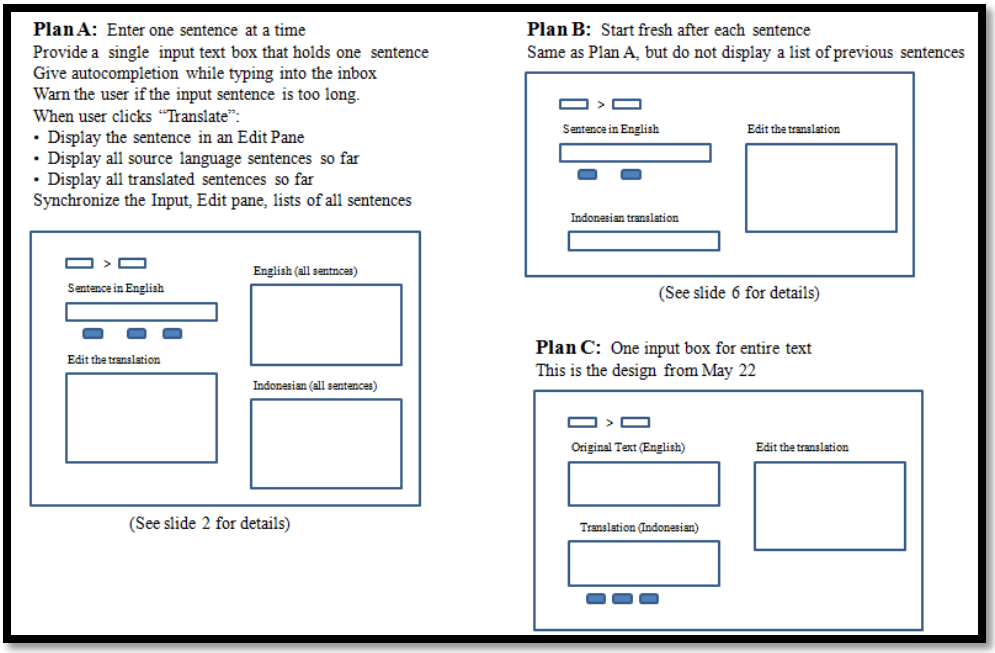


Figure 11: A compromise design supporting freeform input within a sentence and showing full feedback for the sentence

To support real time interactivity, we computed the translation and top two backtranslations for every word in a language pair offline and cached the results in an indexed flat table. We also removed the autocompletion feature while caching translations and disambiguations on client browsers. This served the further purpose of remembering people's disambiguation decisions when they use the word again in the future.

The UI was also modularized so we could easily test out various designs. We isolated the key components of the website and initially laid them out as shown below.

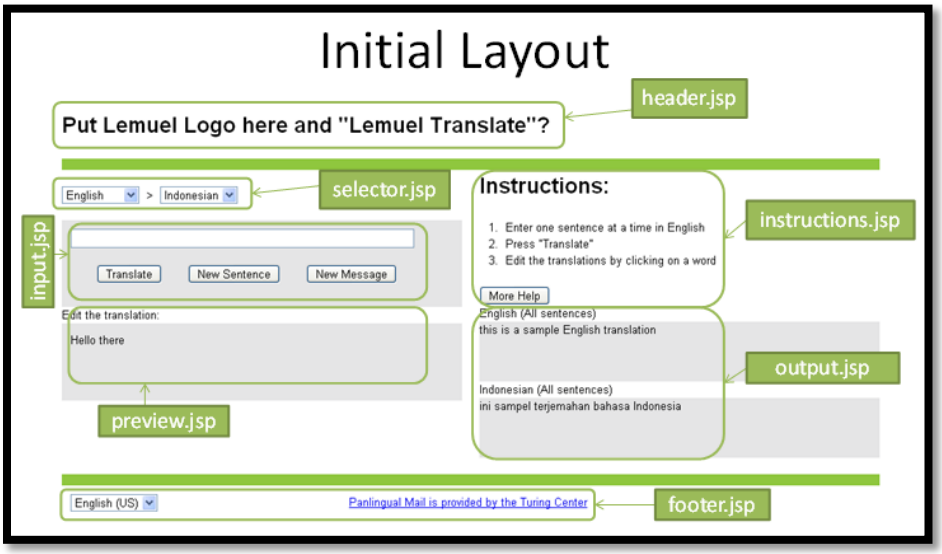


Figure 12: A labeled screenshot of the key components of the Panlingual Translator UI

Figure 13: A screenshot of Panlingual Translator Alpha

Because of our belief in the importance of lemmatic encoding, we chose to promote it by automatically tokenizing recognized words as people typed. This gave live feedback on whether or not words were translatable through the PanDictionary making a highly interactive experience. However, we discovered in internal user testing that people were confused about the unique textbox and preferred a predictable standard one without the automatic tokenization. They felt like the preview panel was confusing and overloaded and that some interactions threw them off. For example, when they click “New Sentence”, all the information in the preview panel and the input widget disappear and the text of their source message and translation appear in the two output boxes on the right. Users thought they made a mistake and had no way of going back.

The advantage of the system was the density of information we could show in the preview panel and the instant feedback we gave users as they typed. Some additional interesting technical features of this prototype include an elegant database-backed localization system that allowed people to contribute interface translations while maintaining a history of contributions and support for corner-case input issues like capturing phrases from the free input textbox, dealing with special characters like numbers and punctuation, and detecting and automatically processing text pasted into the input box. At this point the token dropdowns gave feedback in the following format: “[source word suggestions]: [translation] ([backtranslation1], [backtranslation2])”, which users found very ambiguous without explanation or documentation.

Our Final Attempt: Panlingual Translator Beta

In order to solve the problems with our second attempt, we designed two new powerpoint prototypes as shown below.

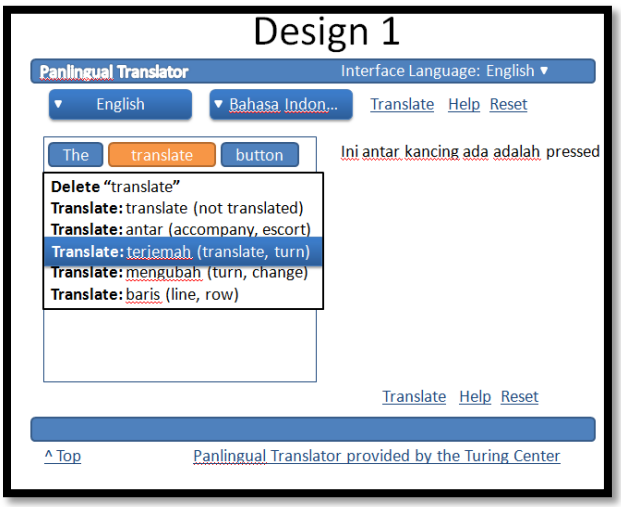


Figure 14: A mockup of a design that tried to balance the amount of information presented

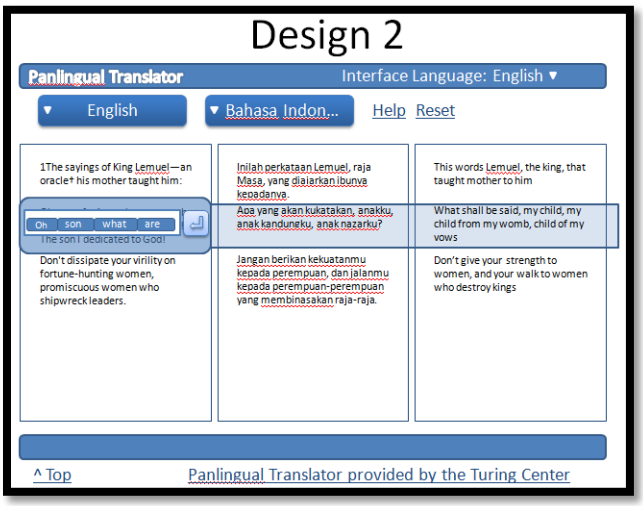


Figure 15: A second design that tried to show the user backtranslation feedback without overwhelming them

The first design gave users a free input box, which tokenized and translated their message after they pressed "Translate". No autocompletions were provided because they were not reliable and returned results in lexicographic order while people seek synonyms and paraphrases of what they are trying to say. After pressing "Translate", problematic tokens would be highlighted so users could correct them. We reduced the information density by not showing backtranslation information and relied instead on an intelligent algorithm for flagging problematic words.

The second design tried to incorporate backtranslation feedback by adding a third column so people could read through the backtranslation and identify problems in the translation based on it. We again tried to enforce lemmatic encoding by making a flexible, but single line input box. This box could scroll to any part of the message enabling that sentence to be edited by revealing its structure in clickable tokens that provided dropdown options to disambiguate the token or the ability to edit tokens and the whole sentence.

We ended up taking ideas from both of these designs and made a third one that combined the simple, free text input of design 1 with the 3-column layout of design 2. The result, which seemed to provide the right balance between showing too much or too little information, is shown below.

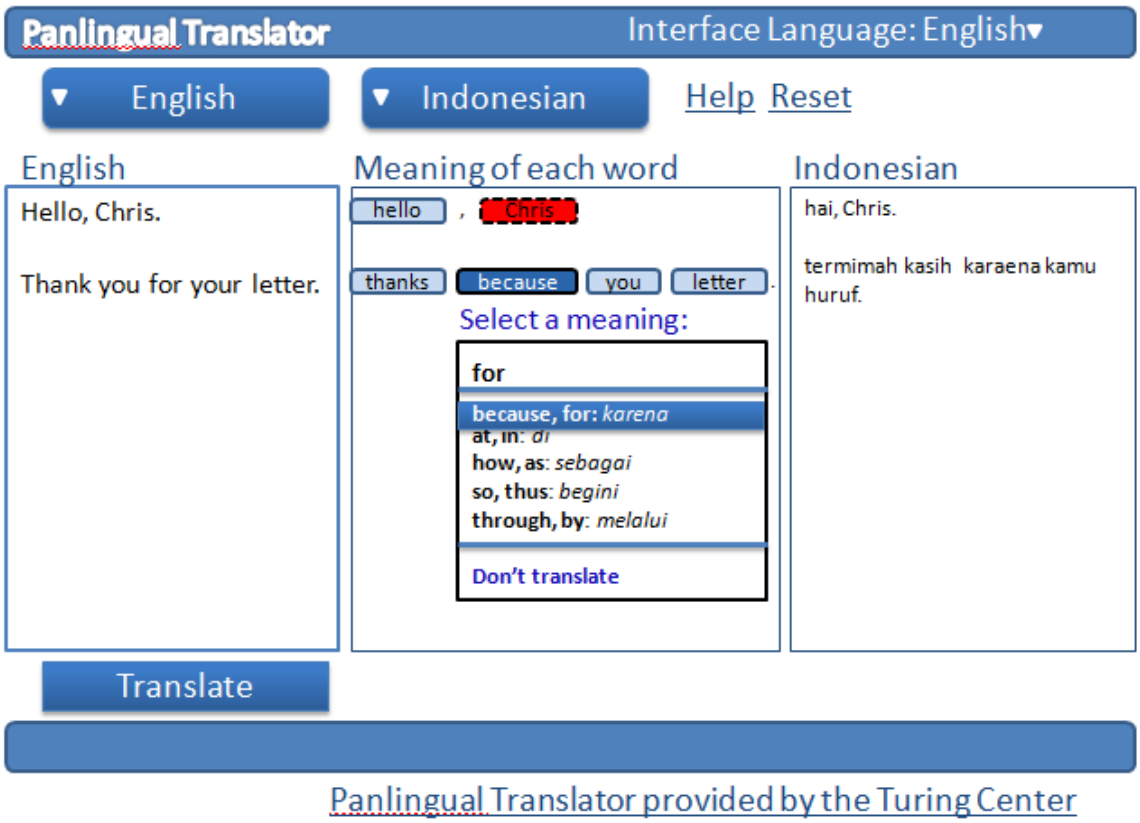


Figure 16: A compromise design combining the free text input with the 3-column aspects of the two previous designs

Now that we decided on backtranslation feedback and "solved" the presentation problem, we focused on creating a usable, intuitive, and learnable user interaction. With our final design we wanted to do three things:

1. **Set user's expectations**. People expect the system to perform fluently and to do so without any additional effort beyond inputting their message. As described in the section on Lemmatic Translation, we designed interacting with Panlingual Translator to be like interacting with a child with a small vocabulary. Thus, people should expect to write simply and concisely, using common words and interactively ensuring the computer understood them. We attempted to do so by making all interface text in the system written in the first person with phrases like "I can't find your word in my dictionary". We also considered adding an avatar-like character to represent the system, but decided not to do this because it took up screen real estate and we did not have time to test the value of adding it.
2. **Support Experimentation** to help people explore the lexicon. We made it easy for users to try out alternative words (e.g. synonyms or paraphrases) without messing up their message or losing their work (see bottom of the dropdown in Figure 17). We found this risk-free interactive model to be a superior way to discover the contents of the PanDictionary than offering long lists of autocompletions. Users were able to quickly find satisfactory ways to express themselves.

3. **Reduce confusion** by labeling everything. The lack of explanatory text made previous interfaces confusing which made users feel lost and unsure about how to use the interface. Questions like "Are backtranslations simplified English?", "What is the middle panel for if I just need a translation?", and "I didn't know I could click on this…" frequently came up in user tests. We decided to add inline documentation for every key user interface element to solve this.

A screenshot of our first implementation is below.

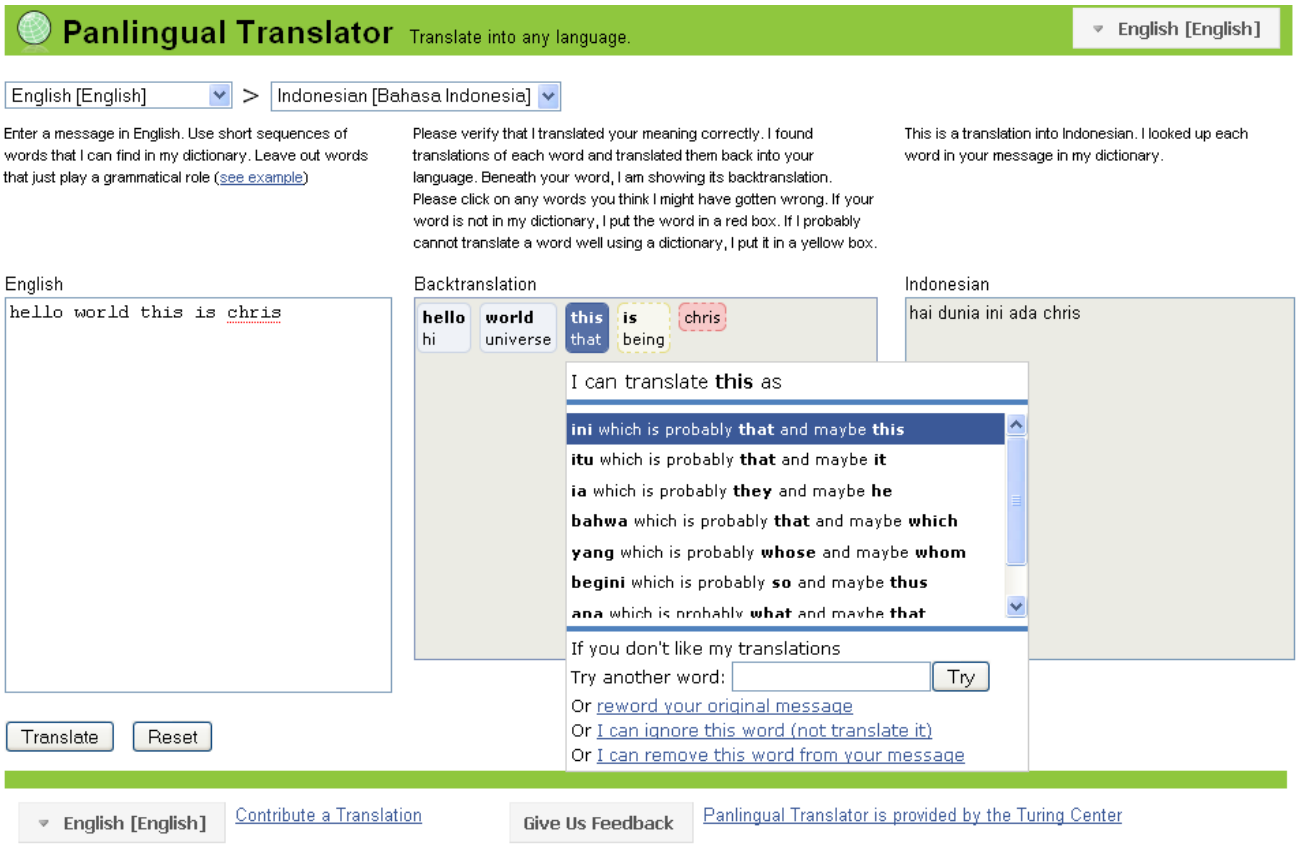


Figure 17: A screenshot of our attempt to explain and document the interface

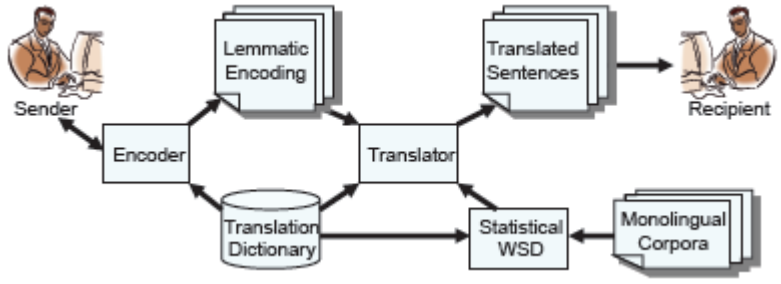


Figure 18: The user-guided architecture for user-guided translation

After implementing the new design, we began more extensive user testing within our larger network of friends and students. We consistently received feedback in several areas:

- **Grammaticality**: Everyone was disappointed with the non-grammatical output because they were testing between languages they knew. We were trying to design a system for people who needed to communicate in minority languages they do not know. We informally asked several people whether or not they could understand the meaning conveyed in the non-grammatical output and answers varied with some people reporting yes and others reporting no.
- **Input**: When people tested between languages they knew, they often wanted to correct the system and contribute a better way to translate. It would have been helpful to take advantage of this desire like Google Translate does. The most common requests include:
    - The ability to do basic affixation and the encoding of basic morphological rules
    - Adding correct translations or new ones to the dictionary
    - Adding part of speech information and control word forms.
- **Feedback**: People loved the backtranslation feedback feature because it made the translation system transparent—they felt like they were in control and knew what was going on. In fact, users wanted more control over the translation output through annotating some linguistic features (e.g. tense) of their message to ensure they are conveyed correctly. Some also wanted to know about the target language (e.g. SVO order) in order to craft good messages. Unfortunately, some people did not realize that backtranslations provide implicit disambiguation—they thought they were explicitly selecting senses.[12]
- **Confidence**: Because of the source language feedback and disambiguation alternatives, people were less uncertain about their translations than SMT translations and confident in their assessments of translation adequacy.
- **Speed**: People love speed and interactivity.
- **Mobile**: Some people tried the website on iPhones, Android phones and BlackBerries, which was obviously cumbersome, since we did not design it for such scenarios. We did receive ideas on reducing the information presented since some of it was not useful.
- **Tokens**: Since two backtranslations are needed to provide an implicit disambiguation, we showed the two highest ranked backtranslations to users when they clicked on a token and the best one on the surface. However, if the highest ranked backtranslation was equivalent to the source word, we showed the second best backtranslation on the surface to prevent users from assuming the word was properly translated (the source word backtranslation is compelling, but retains all the ambiguities of the original word they input). This inconsistent ordering confused people because they assumed the displayed backtranslation was always the best one.
- **Panel Order**: Naïve users initially found that having the backtranslation panel on the far right more intuitive, but it then appeared optional to them and they did not realize they could click on tokens. We then put the backtranslation panel in the center where users would encounter it and deal with it before using their translation, which partially solved the problem. After becoming familiar with the system, users also liked this option better because it clarified the correspondence of tokens with words in the input panel.

[12] Backtranslations are selected using the AllSenses algorithm described earlier, which sums probabilities of words over all possible senses. Choosing a word with two particular backtranslations does not explicitly disambiguate (we haven't mapped it to a single sense), but identifies a set of dominant senses the word belongs to.

- **Documentation**: Almost no one read the documentation. Even when users got stuck they did not read documentation. Some people liked the example message though.

Based on this feedback, we reduced the amount of inline documentation on the website, hid lengthy explanations behind "see more" links, put the source word and backtranslation together in each token to clarify correspondences, added a stopword list to highlight words we knew were typically hard to translate, and continually emphasized that non-grammatical input enables adequate non-grammatical translation for all languages. Our primary remaining task was to teach people how to use the system.

We originally created a tutorial that walked someone through the Panlingual Translator, but discovered that people would simply opt out. We next tried putting a lot of inline explanations as shown in Figure 17, but again people would not read it. We ended up deciding on a 3-part method of teaching: a first time user sees a video[13] (Figure 19), followed by contextual tips (

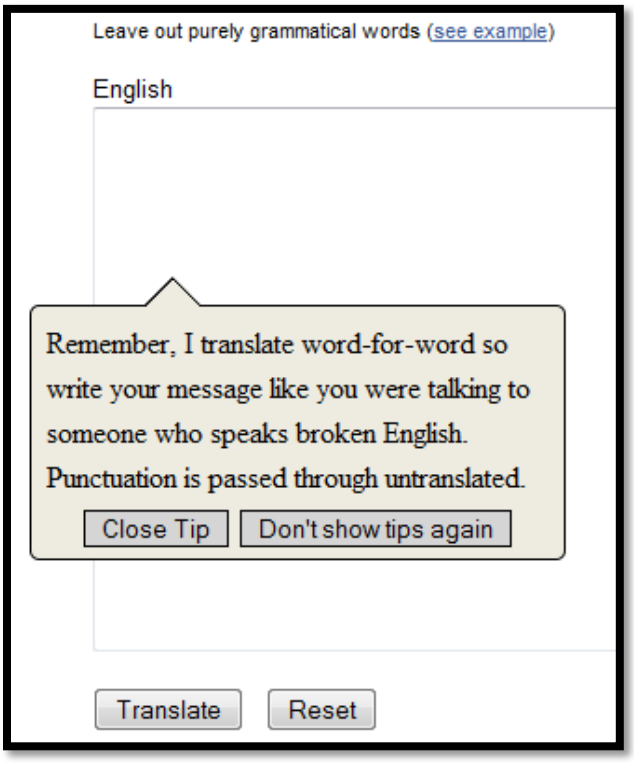


Figure 20), reinforced with short inline explanations (Figure 21). First-time users see copytext that succinctly answers the most important questions and invites them to watch the video. We give clear time limits so they know what they are "buying into" when they click on the video.

[13] The video, which explains the motivations for the system and provides a walkthrough of the system can be seen at http://www.youtube.com/watch?v=AuhfO2iCdyU

Figure 19: What a user would see the first time they visited the Panlingual Translator. We explain the most important points about the system in text and show a video walkthrough.

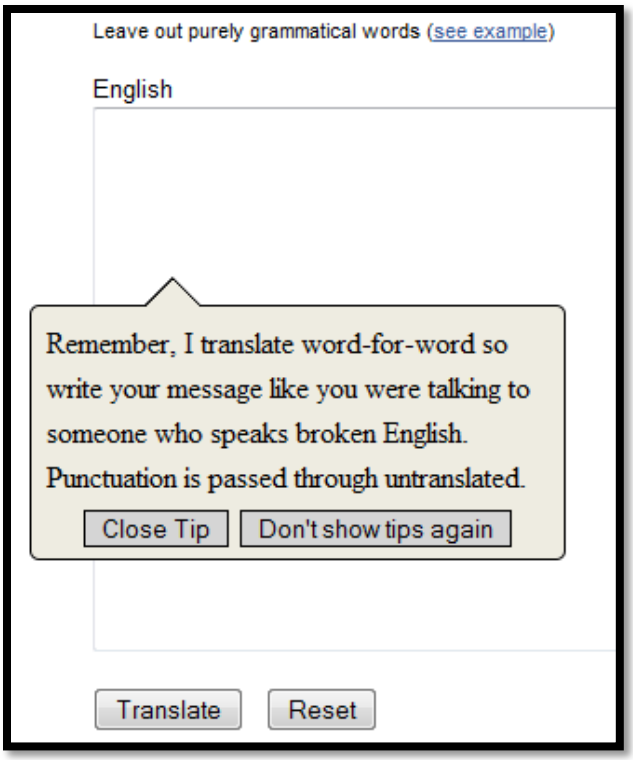


Figure 20: First time users see tips as they go through the translation process to guide them.

After watching the video or quitting early if they get bored, users see tips (

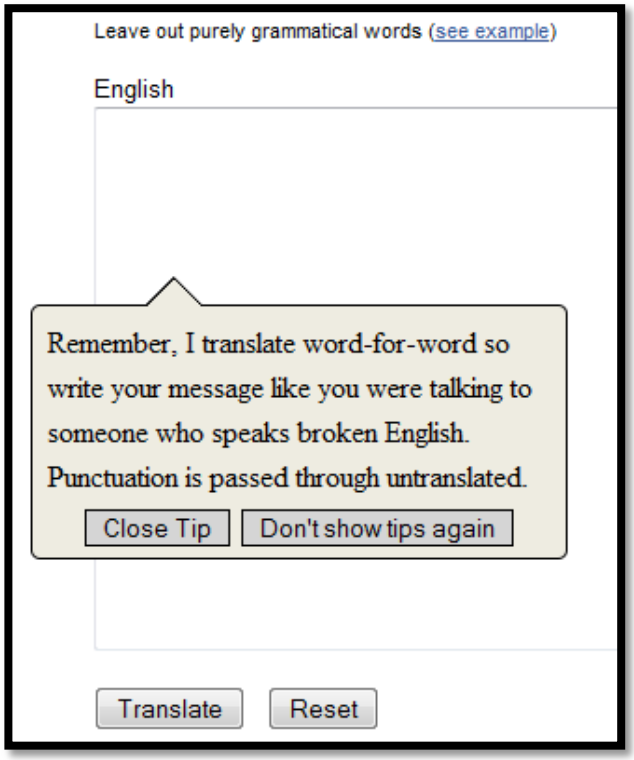


Figure 20). In this example, we re-emphasize the need for lemmaticality to get adequate translation. Since we no longer enforce lemmatic encoding through the design, but let users input freely, we try to compensate through clear documentation. While users we studied did not find this tip system annoying, only about half of our study participants read the actual messages. These tips are only displayed for first time users, but can be re-enabled through a clear help link on the website (which none of our study participants clicked on, even when confused or stuck).

Lastly, we display shortened explanations above each panel with more information available through a link if users want details as shown in Figure 21. Unfortunately, we discovered that most people ignored even the shortened explanations.

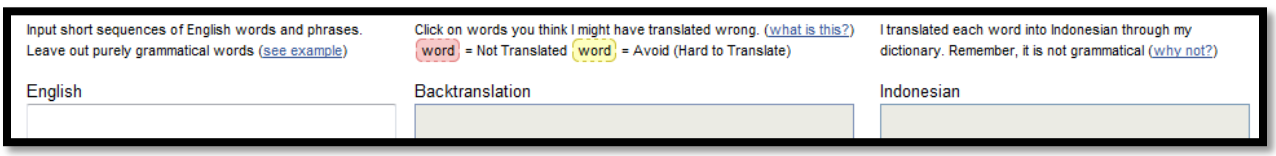


Figure 21: We shortened inline explanations to the most important information and added links that users could click to see more complete explanations.

Overall, it is quite apparent that users do not like learning things up front, but prefer exploring the website and figuring things out for themselves. Many recommended that we show them help only

when they need it by detecting problematic usage of the system (e.g. a lot of untranslatable words or long sentences) and then by responding with a list of recommendations. Users did not mind learning some things up front as long as they were immediately relevant (e.g. "use simple speech").

A screenshot of the current website is shown in Figure 22[14]. With our design and implementation finalized, we conducted an informal user study to evaluate how well we trained non-expert users, how well they could convey meaning using LMT and the reasons why LMT works at all. The next section describes our experiment and results.

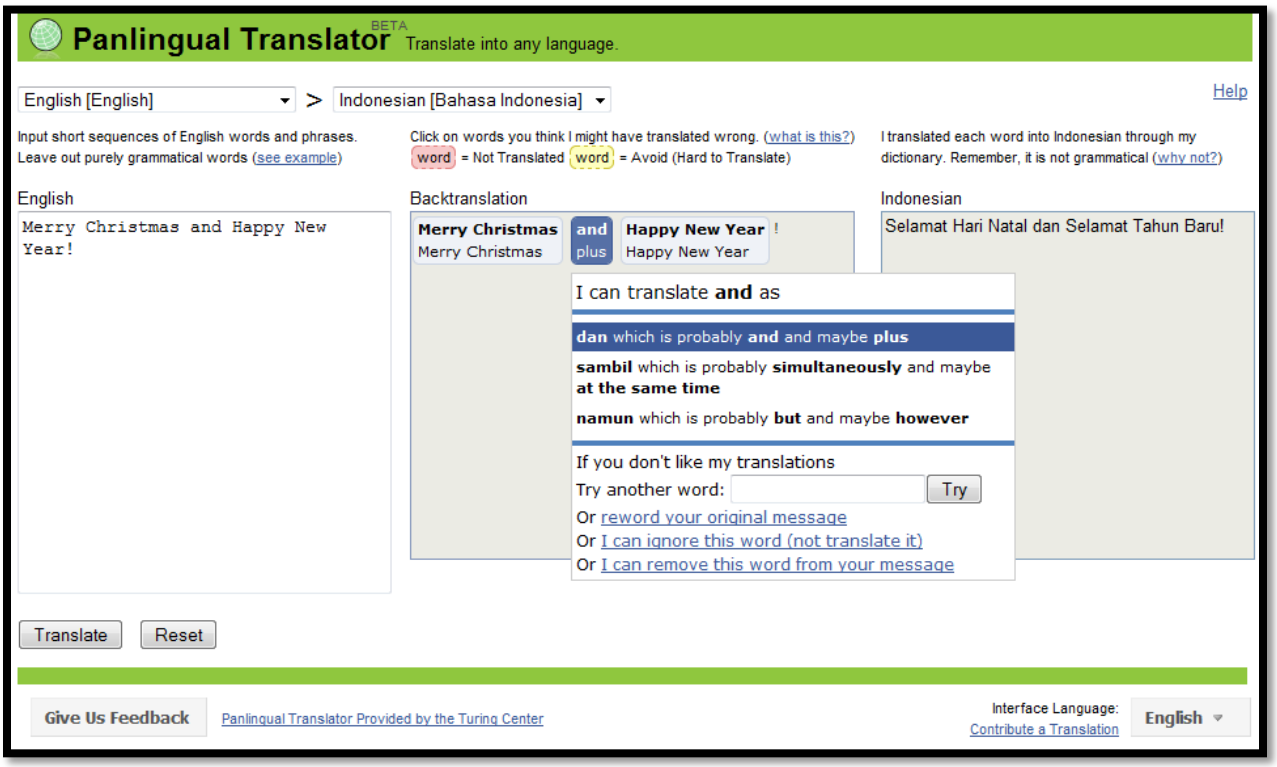


Figure 22: A screenshot of Panlingual Translator Beta

Experimental Results & Analysis

Research Questions

We sought to answer the following questions:

1. How well did we train people to communicate lemmatically?
2. How well do their translations convey their meaning?
3. What is the relationship between a user's level of lemmaticality and the successful interpretation of their translation?

[14] Several other interesting technical aspects of Panlingual Translator not described in this thesis include: a flexible architecture that supports language specific modules, elegant crowdsourced localization, instrumentation, and the inline tip system.

4. What is the relationship between a user's confidence in a translation/interpretation and the actual success in conveying meaning?
5. Why does LMT ever successfully convey meaning?

Dimensions we did not measure include message encoding/decoding[15] time and information density (how informative translations were given their length). We did measure encoder/decoder confidence, decoding difficulty, translation adequacy, and message lemmaticality.

Hypotheses

In response to these questions, we made the following hypotheses:

1. Our Training Video, Introductory Text, and Tips adequately train users to expect non-grammatical, but sufficient translations, and to lemmatically encode messages.
2. The translation adequately conveys people's meaning.
3. More lemmatic messages convey meaning better than non-lemmatic ones.
4. Regardless of lemmaticality, senders and receivers have a good correlation between their confidence in a translation/interpretation and the actual success of meaning conveyance.
5. LMT successfully conveys meaning because short simple sentences are easy to disambiguate given sufficient extra-lingual context.

Setup

To test our hypotheses, we conducted a user test with 12 people selected from a pool of non-Indonesian friends at the University of Washington. We chose the English-Indonesian language pair because Indonesian represents a language with borderline coverage in the PanDictionary, giving us a good baseline on performance. We also had access to a lot of bilingual speakers who could interpret Indonesian messages.

Participants came in knowing that they were about to test a translation system with one having previous knowledge of the PanImages website. Each person saw the Panlingual Translator as a first time user would see it—with the background dimmed and an introductory video in the foreground and brief explanatory text about the website above (see Figure 19). Users were asked what they would do and proceeded accordingly. If the user would watch the introductory video, they clicked play, closing it whenever they no longer wanted to watch, and if they preferred directly using the system, they closed the video and proceeded with the first task.

The tasks were as follows:

- **Teaching:** "Teach someone how to use a Microwave. Translate your message from English into Indonesian. Explain to them what it is used for and how to use it."
- **Reading:** "Read this message and tell me what you think it means. How confident are you that your interpretation is what the author intended?"

[15] Encoders are messages senders and decoders are message recipients.

- **Asking & Telling:** "It's around Christmastime and you want to send a card to your relatives who don't speak your language. Write a short note describing how your family is doing and ask questions about your relatives."

Each study took between 30 minutes and 1 hour to complete. After the first and last task, we asked participants to rate the expected translation adequacy of their message (how well they thought their meaning was conveyed) on a scale of 1 to 5 (5 = perfect meaning conveyance, 1 = no conveyance). We also asked them to rate their confidence in their rating of expected translation adequacy (5 = complete confidence, 1 = no confidence).

For the reading task, we used the Panlingual Translator to get an English translation of an Indonesian news article, which we lemmatically encoded. With only this information as context, we asked participants to write their interpretation of the message and to rate their confidence in their interpretation (5 = complete confidence, 1 = no confidence). We later evaluated how well they interpreted the meaning (5 = perfect interpretation, 1 = completely wrong interpretation).

The Asking & Telling task was evaluated similarly to the Teaching task, except that we had one participant compose a message in French while providing us with a fluent English translation of their message so we could evaluate the translation adequacy of their message.

Once the three tasks were complete, we asked participants for feedback on the Panlingual Translator website. We also had them rate its usefulness and enjoyability on a scale of 1 to 5 with 5 being very useful or very enjoyable and 1 being useless or painful respectively.

Evaluation

After collecting the user study results, we had three bilingual Indonesian speakers write out English interpretations of the translated messages. We also had them rate their confidence in their interpretation for each message along with how difficult it was to interpret (1 = very easy, 5 = very difficult). We then compared their interpretations sentence by sentence with the source messages and rated the correctness[16] of each interpretation (5 = perfect interpretation, 1 = completely wrong interpretation). We also rated the Lemmaticality of each source sentence where Lemmaticality is a subjective measure based on sentence length, simplicity of word forms, and non-grammaticality[17].

If our system enables non-expert users to confidently send and receive their main meanings in a message translated into a low coverage language like Indonesian, we believe that the practical utility of LMT will have been demonstrated.

[16] We take the terms "translation adequacy" and "correct interpretation" by a recipient to be synonymous and use them interchangeably.

[17] The criteria include: sentence length, inclusion/exclusion of stoplist words like auxiliary verbs, use of punctuation and use of simple, translatable words.

Results

We had a total of 12 participants, 17 messages and 120 sentences. About two thirds of all sentences were adequately translated[18], which seems to confirm our second hypothesis. Users generally created lemmatic sentences and when they did not, they were intentional about it. However, a more detailed look at the data reveals that the non-lemmatic sentences were largely composed by participants who did not watch the introductory video. The distribution of correctness (translation adequacy) and lemmaticality is shown in Figure 23 and Figure 24 respectively.

It appears that if users follow the provided training they successfully learn to lemmatically encode their messages. Some said they would watch the video and close it after the first minute intro, but often watched the whole thing for the purposes of the user study. This could affect the results, but it seems safe to say that the video and tips taught users how to craft their messages. An open question is how to ensure users go through the training.

We also wanted to know if we adequately taught people how to use the system end-to-end, not just with respect to lemmatic input. The anecdotal evidence suggests that most users can successfully navigate the whole system, confirming our first hypothesis, but there were several problem areas such as confusion over the meaning of backtranslations and users not realizing that tokens can be clicked (despite affordances like underlining text on mouse hovers).

[18] We define adequate translation to be a sentence with a correctness rating >= 4 out of 5

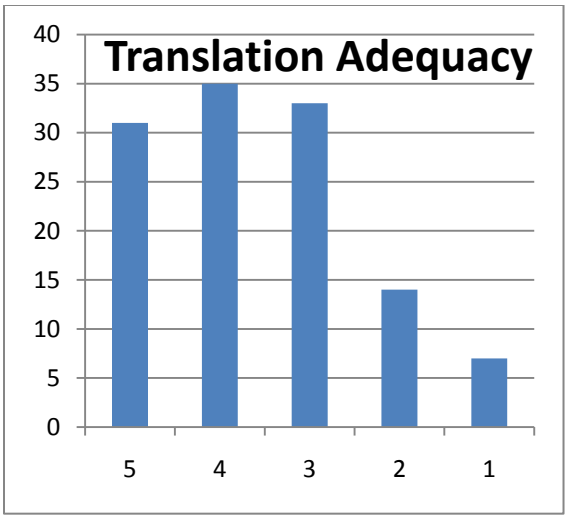


Figure 23: Distribution of sentence translation adequacy. About two thirds had high adequacy (a rating of 4 or 5)

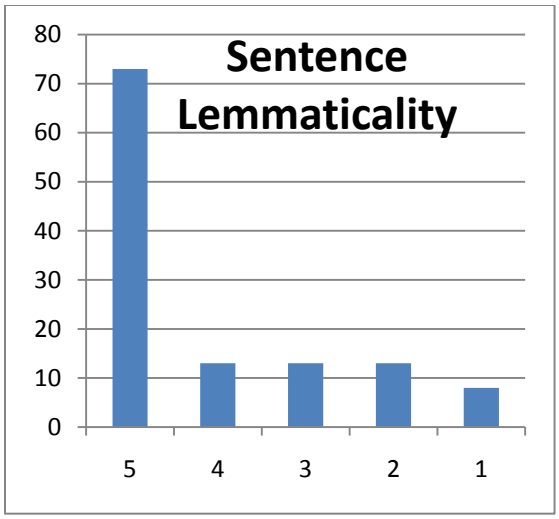


Figure 24: Distribution of sentence lemmaticality. A rating of 5 means the sentence is very lemmatic.

After averaging ratings from our translation interpreters, we calculated a message level correlation matrix (see Table 1). Since encoders and decoders did not rate messages sentence by sentence, we were only able to calculate a sentence level correlation matrix over the dimensions of lemmaticality, length and correctness (see Table 2). None of these correlations is statistically significant except for the expected positive relationship between sentence length and lemmaticality.

Message	Lemmaticality	Length	Difficulty	Confidence	Correctness	SenderGoodness	SenderConfidence
Lemmaticality	1	-0.86425	-0.35943	0.137647	0.108989	-0.19254	0.132491
Length		1	0.46845	-0.15546	-0.05321	0.217374	-0.28572
Difficulty			1	-0.44377	-0.21851	0.101775	-0.41118
Confidence				1	0.489409	0.061517	-0.16349
Correctness					1	0.472155	0.014895
SenderGoodness						1	0.176604
SenderConfidence							1

Table 1: A correlation matrix of message ratings. Only Length and Lemmaticality have a statistically significant correlation, but potentially interesting correlations have been highlighted.

Sentence	Lemmaticality	Length	Correctness
Lemmaticality	1	-0.67938	0.152107
Length		1	-0.03238
Correctness			1

Table 2: Correlation matrix of sentence ratings

The most surprising result from these correlation matrices is that there is no correlation between lemmaticality and correctness, shattering one of our most important assumptions (our third hypothesis) and apparently making all of our attempts to enforce lemmaticality moot! However, there may be some correlation between lemmaticality and interpretive difficulty, which means that lemmatic messages are still preferable. Some other potentially interesting correlations include:

- A positive relationship between the **average sentence length** of a message and **difficulty in interpretation**—longer messages are harder to interpret.
- A negative relationship between **interpretive difficulty** and **confidence in interpretation**—messages that are hard to interpret result in less confidence in the interpretations.
- A positive relationship between **confidence in interpretation** and **correctness**—interpreters are more confident when their interpretations are correct.
- A slight negative relationship between **interpretive difficulty** and **correctness**—recipients may make more mistakes with difficult messages.
- A slight positive relationship between **average sentence length** of a message and **sender's perception of goodness**—senders may feel like they conveyed more of their meaning with longer sentences.
- A slight negative relationship between **average sentence length** of a message and a **sender's confidence** in his or her perception of goodness of a translation—longer sentences may also make senders less confident that it will be translated properly.
- A positive relationship between **correct interpretation** and a **sender's perception of goodness**—this is an excellent result that means when senders believe they have a good translation (due to backtranslation feedback), that translation is often correctly interpreted.
- A negative relationship between **interpretive difficulty** and a **sender's confidence** in his or her perception of the goodness of a translation—this is another good result because senders may be able to tell when recipients will have a hard time interpreting their message, based on their own confidence in a translation.

Overall, these correlations are evidence confirming our fourth hypothesis: regardless of lemmaticality, when users are confident that they have a good translation based on the backtranslation feedback provided, a recipient's interpretation is likely to be correct. Senders can tell when their message is being translated badly or translated well.

At the same time, the surprising non-correlation between lemmaticality and translation adequacy made us wonder: what key features of adequate grammatical and non-grammatical messages make them intelligible and successfully convey meaning? Perhaps the simplicity of lemmatic encoding deals more with the problem of limited linguistic data. Since lemmaticality is a practical term that says, "Use what's in the dictionary", languages with sparse resources may require short, simple sentences, but languages with rich dictionaries do not.[19] We still advocate lemmatic encoding because it appears to help recipients more easily and confidently interpret the sender's meaning.

After discovering that LMT results in translation adequacy with even non-lemmatic source messages (thereby disconfirming our fifth hypothesis), we examined positive examples of lemmatic and non-lemmatic messages in order to understand why LMT works. Table 3 shows such an example (underlined words were passed through untranslated because they were not in our dictionary).

[19] We did not study situations with minority source languages. This is future work that would help to determine the generality of our findings in this study.

		Best Example	No Lemmaticality Example	High Lemmaticality Example
Message		use microwave to heat food depress lower right to open microwave door put food inside microwave close microwave door depress number to select cook time depress "start" to begin cook hear microwave scream when end depress lower right to open microwave door again withdraw food enjoy	A microwave is a device that cooks food. To use a microwave, take food and put it on a plate. Open the door of microwave and put the food in. Close the door of microwave. Then press the button by the door to show to how long you want your food cooked. If you push 1,0,0 then cook your food during 60 seconds and if you push 2,0,0 then it cooks your food during 120 seconds, or 2 minutes.	microwave oven heat food pull handle to open door put food inside close door press start button
Average	Lemmaticality	4	1	5
	Length	4.6	13	3.4
	Difficulty	3.333	4	2.667
	Adequacy	4.6	3.9722	3.933

Table 3: A comparison of three adequately translated messages with varying lemmaticality.

The middle column shows a message without lemmatic encoding that achieved an adequacy of almost 4 compared with a very lemmatic message in the right column that achieved a similar adequacy. Clearly there is a difference in the amount of information being conveyed, which we did not account for in this study, but the fact that both types of messages attain similar translation adequacy is striking. In the far left column, we see a message that falls in between the first two in terms of both lemmaticality and information density. This message in fact achieved the highest translation adequacy of all messages with a score of 4.6. The non-lemmatic message was the most difficult to interpret with a difficulty score of 4, while the highly lemmatic one was the easiest with a score of 2.667.

Unlike Soderland et al.'s original work (3), our system used primarily high-probability translations, so we cannot directly discover the effect of mistranslations via low probability words (we must manually examine each word for mistranslations[20]). Even so, we believe anecdotal evidence shows that mistranslations are a major source of misinterpretation and conversely that correct lexical translation is the important factor in correct interpretation.

For example, one obvious difference between the two messages is the fact that the verbose, non-lemmatic message has a significant number of untranslatable words. Bilingual interpreters read these words without any problems and are then able to interpret the whole message because they are grounded in some key terminology. On the other hand, the lemmatic message, because it was so concise had very little room for confusion (a two word sentence is hard to get wrong), so getting a few keywords like "door" translated correctly enables the interpreters to infer that the probable action being referred to is "pull the door open" even if the previous words were mistranslated. We conjecture that it may not be the common surface features of lemmatic and non-lemmatic messages that make them equally adequate, but the more general feature of unambiguous terminology, properly translated.

[20] Native Indonesian speakers can indicate particular words that were clearly mistranslated, but this level of detail and effort was beyond the scope of this study.

It might seem that lemmatic messages are more likely to have properly translated unambiguous terminology, especially in cases of one or two word sentences. However, the non-correlation of lemmaticality and correctness indicates that some lemmatic sentences are misinterpreted. The risk in short sentences is that if any of the words is mistranslated, there is little additional context by which a recipient can derive a correct interpretation.

For example, one user had a sentence "bye", which was translated into Indonesian as "hai" ("hello" in English), a word that does not denote "bye" at all. The backtranslation feedback revealed this, but the user thought Indonesian might be a language that used the same word for greetings and farewells. This was consistently misinterpreted by recipients, showing that the surrounding context of the message was not sufficient to correct it. The single word also made interpreters more confident in their interpretation when it was actually wrong.

A second example is the sentence "this heat food 1 minute", translated "ini panas pangan scientific term 1 baik"[21], interpreted as "heat one well" simply because the last word "baik" is a mistranslation of minute. The backtranslations of "baik" were "nice" and "good", so this may be a situation of user error, but it shows how vulnerable lemmatic sentences can be to mistranslation. On the other hand, long sentences like the one immediately following the previous example, "food not hot repeat the action, it become more warm each time", translated as "pangan scientific term tak panas mengulang ini buat, dia jadi lagi panas setiap kali", with a backtranslation of "meal no warm reiterate that act, they get again hot every occasion", was consistently correctly interpreted as, "if food isn't hot, repeat because it'll get hotter each time". Although this sentence is long, it is lemmatic (reduces extraneous words) and every word was translated well. Since lexical translation depends on lexical semantics, it is no wonder that mistranslations are a major reason why meaning is not successfully conveyed.

In order to understand when and why LMT works, we also studied the particularly badly translated sentences shown below:

Source	Correct Interpretation	One Interpretation	Problem
friend-friend at DPR, whose at floor 21 everyone get off towards floor one, say Pakaya.	"Friends at DPR, who were at the 21st floor all went down to the first floor", said Pakaya.	Everyone at DPR, which was on the 21st floor, moved to the first floor, said Pakaya. *Not sure about first part*.	Ambiguity of "friend-friend". Could be "friend of a friend" or "close friend"
center earthquake at strait Sunda 42 kilometer northwest Ujung Kulon to inside 10 kilometer.	The epicenter of the earthquake was at the Strait of Sunda, 42km northwest of Ujung Kulon at a depth of 10 km.	The epicenter of the earthquake was in the strait of Sunda 42 kilometers northwest of Ujung Kulon, and the strait is 10 kilometers deep.	Ambiguity of "inside". Possibilities: within an area or radius of 10km, etc.

[21] The words "scientific term" are residual noise from our dictionary processing code, which neglected to strip out undistinguished metadata from the unstructured wiktionaries at wiktionary.com. For this experiment, we explained to the interpreters the reason for the bug and told them to ignore it.

yesterday evening we reach record low temperature	We reached a record low temperature last night	We completed the report yesterday afternoon	Record was mistranslated as "report" causing interpreters to ignore temperature, which was mistranslated as "fever".
happy christmas, hope you well.	Merry Christmas, hope you are well.	Enjoying Christmas, I hope you are fine.	"happy" was mistranslated as "satisfied" preventing the common idiomatic interpretation of Merry Christmas from being transmitted.

Table 4: A table of four sentences that were consistently misinterpreted.

In each of these examples, there is either a difficult ambiguity or a mistranslated word, which caused the error in interpretation. Clearly, the importance of the error varies according to situation, but these examples reveal that selecting accurate, unambiguous translations for key terminology is very important. We may be able to adapt work from Christensen et al. (15) on finding good terminology for image search queries by using their criteria to algorithmically find good translations.

Yet, even with the present system, looking at the backtranslations reveals that users were given sufficient feedback to notice the translation problems. For instance, the backtranslation of the third example reads, "yesterday afternoon us achieve report inferior fever". This would probably be misinterpreted by even a native English speaker! Instead of leaving the word "record" (backtranslation: "report") alone, the user could have overcome the dictionary limitations of not having a translation in the right sense by rephrasing his sentence into something like "yesterday evening we reach most cold in decade", which also resolves the problem with the mistranslation of "temperature". Motivated users of our current system should be able to detect and respond to these problems.

Lastly, a lot of translation issues were resolved through domain knowledge. When interpreting the Indonesian news article, users said, "A lot of this comes from familiarity with news stories and how they're worded" and many of the disambiguations concerning the meanings of various numbers was through domain knowledge about earthquakes. Indeed, when people lacked domain knowledge, even correct translation was insufficient for complete meaning transfer (e.g. someone didn't know what "lofty" meant and interpreted it as "weak", which affected their interpretation of the whole sentence to be that "the earthquake made all buildings in Jakarta weak" instead of the meaning that "all tall buildings swayed").

Other Results

Usefulness	Enjoyability	Reading Confidence

3.2778	4.125	3.9433

Table 5: Miscellaneous metrics from the user study.
Most participants enjoyed using the system and found it rather useful.
They also were confident in their interpretations of the Indonesian news article.

Why users were confident and enjoyed using Panlingual Translator

When questioned about why they confidently thought their translations or interpretations were good, several users mentioned the criteria of **coherence**, "my interpretation makes sense", **comprehensive exploration**, "[because of] the exploration of the meanings and words and being able to use the drop down list, define alternative words" and **iterative development**, "I liked how you could choose words and other translations; it was really good to have instant feedback so you could iteratively improve what you were sending". Indeed some said they "loved" the backtranslation feedback and control Panlingual Translator gave them over their output. They also liked the interactive speed and visually appealing layout, color scheme and design.

Many said they enjoyed using Panlingual Translator (several said it was actually fun) because it felt like a puzzle or a game. They found thinking of synonyms and paraphrases fun and could imagine doing so with friends for fun. Others said they took pleasure in discovering, through translation, the surprising ambiguities of words they took for granted. Anecdotally speaking, it seems like verbally-minded people[22] enjoy using and are more capable of using the system than less verbally-minded people because it is easier for them to find synonyms or rephrasings and they enjoy learning about the target language and the challenge of crafting well-translated messages.

Why Lemmatic Machine Translation Works

From anecdotal examples and user feedback, we have come to believe that LMT works because of shared domain knowledge. When a sender and recipient have correctly identified the shared domain they are operating in, the loss in meaning possibility, precision and information density inherent in lemmatic communication are overcome. Furthermore, interpreters can even recover from mistranslations or missing information using the criteria of interpretive coherence. This is consistent with hermeneutical theory (16), which uses the language of genres, or sets of related meanings, in place of domain knowledge.

Lemmatic translation[23] is the means by which two conversation partners without a shared lexicon can explore each other's lexicons and identify the domain or intrinsic genre of their conversation.[24] Using relatively unambiguous correct translations of key terminology, a recipient can identify the intrinsic genre of a sender's message and then disambiguate other words using the relevant word-meanings associated with that particular genre. When an accurate shared lexicon has been established, new words can be coined and old words can be used in novel ways as evidenced by people's defining the untranslatable term "microwave" in terms of translatable words.

---

[22] Verbally minded people probably enjoy word games like crosswords and like learning languages and writing.
[23] There are other means to identify shared domain such as by grounding in images, pointing to objects, or acting.
[24] See Footnote 8 for details on the interpretive process.

Thus, when enough meaning has been correctly transferred through LMT, a recipient re-cognizes and reconstructs the sender's meaning and, having captured the essential communication, is able to interpret the whole message. This leads to a new important question for future research: "How do we identify the key terminology that must be correctly translated and how do we know when enough has been correctly translated to adequately convey the intrinsic genre of the message?" The answer to this question might reveal the limitations of LMT and provide useful feedback for users.

Implications and Best Practices

The implications of our theory and empirical results on creating messages for LMT include the following best practices:

- **Ensure Native Intelligibility:** Sometimes a source message gets mangled because people are so intent on finding words that work and when that happens, the output is bad because the input became bad (i.e. a native speaker of the source language message would find it ambiguous/hard to understand). For example, "You have love holiday" was consistently misinterpreted (it was supposed to be something like "Enjoy your holiday") and is difficult to interpret even in its native English. Generally, a message must be intelligible in its source language in order for it to be adequately translated. An intelligible backtranslation seems to increase the likelihood of translation adequacy even more.
- **Select Robust Meaning:** Prefer words that are most difficult to mistranslate because of the importance of correctly translating a minimal set of key terminology. Generally, this includes concrete words and idioms. When idioms like "how are you doing" are translatable, it is much better to use them than trying to frame them in novel ways like "are you doing alright?" In cases where there are no good options, use several words that make sense together under a relatively unambiguous interpretation such as "open door / pull handle or knob". Define untranslatable words in terms of translatable ones to ensure consistent interpretation of the word and simplify future usage.
- **Use Shared Domain Knowledge:** Understand the target audience and write messages at their level of knowledge. Build up shared concepts using words that are confidently translated and ideas known to the recipient. Do not over explain as this may cause more confusion.
- **Prefer Lemmaticality:** Although this was not correlated with meaning correctness, it did seem to ease the burden on decoders when interpreting messages.

Conclusion

We set out to build a system that enables anyone to practically translate into any language in the world using Lemmatic Machine Translation. This scalable approach only requires bilingual dictionaries and user cooperation to produce adequate translations. After iterating through three prototypes, we evaluated an implementation of our final product, called Panlingual Translator, through an informal user study. This study revealed that first-time users were adequately trained to use our system and successfully conveyed messages they composed in English to Indonesian speakers. They found the system easy and fun to use, some going as far as saying that it felt like a game.

The study also disconfirmed our assumption that lemmatic messages would be more adequately translated than non-lemmatic ones. We did however discover a correlation and consistency between user confidence and translation adequacy, which means that a good backtranslation implies a good translation. The more confident users are in the translation of their message, the higher the translation adequacy of their message.

A more in-depth investigation of lemmatic messages led us to conjecture that the most important factors in determining translation adequacy are the level of shared domain knowledge between sender and recipient, the intelligibility of the source text, and accurate, unambiguous translation of key terminology. In particular, we discovered that adequate translation is highly dependent on domain knowledge which sets people's meaning expectations in a way that enables them to correctly interpret even mistranslated words. Without sufficient shared domain knowledge, ambiguous lexical streams are either misinterpreted or unintelligible, and thereby ignored. User motivation is also a very critical factor; using Panlingual Translator's tools to interactively develop an intelligible backtranslation corresponded with adequately translated messages.

Having described the design, implementation and evaluation of Panlingual Translator, we have demonstrated that LMT enables non-expert users to adequately translate into even low coverage languages, which suggests that it may support practical translation between all languages in the world. We conclude with a list of lessons learned and directions for future work.

Lessons Learned

Apart from investigating the phenomenon of adequate lemmatic communication, we also learned several important lessons about people and engineering such as:

- People want to experiment without risk by being able to easily undo/redo
- People do not like being forced to deal with problems, but appreciate feedback that makes them aware of problems and having the tools to solve them (and control their translations)
- People do not like learning about a system until the knowledge is relevant. They want to discover interfaces for themselves; they want to learn as they go and are motivated to read text or watch a video only when they encounter a situation where they need it. Discoverable interfaces and contextual help are superior to extensive tips, labels, and other documentation.
- People tend to only use the most basic features. This is similar to the previous point because advanced features take time and effort to learn while known features may seem good enough.
- Intuitive user designs are logical. The interface should provide answers to the questions users are asking at each stage of their task (e.g. "What can I do with a translation I don't like?")
- Do not underestimate the value of interactive speeds. Responsive applications are more usable and enjoyable. Slow ones are painful and not useful.
- Translation does not have to be a mere utility, but can also be a form of entertainment. People had fun using the system and felt like it was a puzzle or a word game. In fact, many said they would use it primarily for entertainment value.
- Modular code is not only easier to maintain, but also makes prototyping alternative user interface designs easier.

- Creating Powerpoint prototypes of UI designs is a quick and flexible way to concretely communicate ideas.
- Accelerating applications early by implementing features like offline processing, caching and adding database indices have huge productivity returns because of the increased interactivity and turnaround time when testing and using the software.
- Using good libraries and frameworks take time to learn up front, but can significantly improve code and accelerate development because they embody best practices.
- It is better to underestimate how much one can accomplish in a day and plan accordingly than to overestimate.
- Discussing, debating and thinking through ideas and problems with others can be significantly faster than going it alone.
- Learning and reflection are a critical part of the design and implementation process, which should not be neglected out of complacency or in order to meet a deadline. A rushed product that is not thought through is likely to fail without providing any insight or value.

Future Work

Although we got very promising results with the present Panlingual Translator, there are several short term improvements that should be made before it becomes a publicly available website: the code should be optimized and compressed, the crowdsourced localization features need to be fully implemented, the introductory video should be split into several shorter segments, user disambiguated tokens should be allowed to show backtranslations that are the same as their source word and user input should be incorporated so past disambiguations and paraphrasings are remembered for future cases.

Scientifically, a more rigorous evaluation of the system needs to be conducted in more languages, including situations where messages are composed in minority languages. We need to get a larger sample size, test in more languages to discover problems peculiar to language families, test in mobile scenarios and in developing regions where this technology can be most helpful. Field testing can help us precisely determine what tasks Panlingual Translator is useful for and what level of dictionary coverage is needed to be adequately useful. We may also gain more insight into LMT and its limitations.

Algorithmically, we need to develop ways to automatically detect problematic input so users can fix it without having to go through every token. We also need algorithms to ensure a wide coverage of senses in the dropdown lists and to offer synonym/paraphrase suggestions (6) instead of lexicographically ordered “autocompletions”. We can also incorporate the corpus-based WSD algorithm described in Soderland et al. (3) or the unambiguous terminology system described in Christensen et al. (15) to select better default translations without user intervention. Using techniques from active learning to incorporate user feedback and instrumented data (e.g. increasing probabilities as people select certain translations more often than others) might also be helpful.

In terms of engineering, we need to build language specific capabilities like transliterations, tokenization for languages that are not space delimited, and ensuring full Unicode support. We also need to implement a reliable crowdsourced interface translation system and a larger user contribution system

so people can add rules and words to our dictionaries. We could engineer support for basic grammar rules like SOV reordering, a part of speech tagger, and stemmers to offer lemmatic suggestions. Looking more broadly, if we wanted to expand to new platforms, we would need to build an LMT API and improve the reliability, scalability and performance of the backend through better indices, caching and optimized code. One such application would be a mobile version of the Panlingual Translator that can run offline in more useful contexts.

Practically, the best way to increase the utility and translation adequacy of the Panlingual Translator is to increase its dictionary coverage and quality. This can be done through licensing data and partnering with linguistic organizations like SIL International. We can also accelerate the acquisition of basic linguistic data by deploying tools like Open Data Kit or WeSay (17) to empower minority language speakers to participate in dictionary building. These tools may enable us to build a multimedia dictionary with images, audio and video associated with each word/sense, unleashing many compelling types of feedback and user interaction possibilities (e.g. recordings of pronunciations could be stringed together for basic speech synthesis). One open question is how to prioritize which languages to support and how to strategically add language-specific capabilities.

Another way to increase Panlingual Translator's utility may be to convert it into a meta-translation engine for languages supported by machine translation websites like Google Translate, Microsoft Translator and Systran. By accessing their APIs, messages input to the Panlingual Translator can be translated by all three engines (when the needed language pair is supported). Our system can merge the output and add interactive feedback, disambiguation and correction tools on top of the "fluent" output so users get the best from all MT approaches.

From a human computer interaction perspective, we could experiment with new form factors (e.g. mobile), new types of feedback (e.g. images) and different ways of presenting all our available information. We have translated definitions for some words, part of speech tags for most senses, and lots of probability information. Giving users this feedback in a non-overwhelming way may help them confidently compose good messages. We also realize that we need a way for people to rate the usefulness of the Panlingual Translator output and are considering adding a five-star rating feature. Based on our design iterations, we believe the ideal desktop interface a Microsoft Word-like interactive environment where free text input, feedback and editing are all integrated in a single panel. Translation problems could be highlighted with red underlines and corrected through a contextual dropdown menu of alternatives or users could do a "translation check" and step through issues one by one, resolving them as they go. A reliable, robust implementation of such a system was unfortunately beyond our capabilities given our resources and timeframe.

On the business side, Panlingual Translator needs to be adapted to meet particular customer scenario needs and heavily marketed. It needs to be tested in international contexts where no alternative translation systems exist to see if it fulfills its promise of serving the Long Tail of languages in the field. We also need to develop better training, with a shorter, multilingual video, broken down into several segments and intelligent detection of user behavior so we can teach them as they use the system

instead of forcing them to learn everything up front. We may also want to help set user expectations by further humanizing the system (e.g. through a culturally appropriate childlike cartoonish avatar).

Surprisingly, it might also be marketed as a language learning aid or a word game. Many users found it fun and would enjoy doing it with friends just to discover surprising mistranslations and ambiguities of language while solving the communication puzzle. Given a clear objective a metric for success, people can be motivated to compose and iteratively improve messages to accomplish the task. This presents an opportunity for a purposeful game. If there were a set of important tasks or documents that needed translation, this process could be transformed into a competition or a crowdsourced game where individuals take a translation task and figure out the "puzzle" of how to adequately encode it into a target minority language. Eventually a knowledge base of valuable translations, intended for practical use in various contexts would be developed (e.g. teaching rural villages about public health in remote islands of Indonesia).

Acknowledgements

I would like to thank my God and Savior Jesus Christ for the honor of doing this research and the grace to see this thesis through the finish. I would also like to thank my family for their support, insightful questions and listening ears. To the many friends and colleagues who gave feedback on the Panlingual Translator: thanks for all your helpful ideas and critiques! Thanks also to my project sponsor Jonathan Pool and the UW Computer Science Department, which has been such a great place to learn. Last, but certainly not least, I want to thank my supervisor Stephen Soderland and my professor Oren Etzioni for taking me under their wing, guiding me through the research and giving me the opportunity to build on their fascinating work. SDG.

References

1. **Raymond G. Gordon, Jr., [ed.].** *Ethnologue: Languages of the World.* 15. s.l. : SIL International, 2005.

2. *Frontiers in Linguistic Annotation for Lower-Density Languages.* **Maxwell, M. and Hughes, B.** 2006. Proceedings of the COLING/ACL 2006 Workshop on Frontiers in Linguistically Annotated Corpora. Association for Computational Linguistics. pp. 29-37.

3. *Lemmatic Machine Translation.* **Soderland, Stephen, et al.** Ottawa, Canada : s.n., August 2009. Machine Translation Summit XII.

4. *Removing the Distinction Between a Translation Memory, Bilingual Dictionary and a Parallel Corpus.* **Vandehinste, Vincent.** London : s.n., 2007. Proceedings of Translating and the Computer 29 (ASLIB). pp. 279-293.

5. *The present status of automatic translation of languages.* **Bar-Hillel, Yehoshua.** 1, 1960, Advances in Computers, pp. 91-163.

6. *Context-based Machine Translation.* **Carbonell, J., et al.** Cambridge, MA, USA : s.n., 2006. Proceedings of the 7th Conference of the Association for Machine Translation. pp. 19-28.

7. *Survey of the State of the Art in Human Language Technology.* **Cole, Ronald, et al., [ed.].** 1997.

8. *Controlled language and knowledge-based machine translation: Principles and practice.* **Nyberg, E. H. and Mitamura, T.** 1996. Workshop on Controlled Language Applications (CLAW-96).

9. *Lexical translation with application to image search on the Web.* **Etzioni, Oren, et al.** 2007. Machine Translation Summit XI.

10. **Helmreich, S., Guthrie, L. and Wilks, Y.** *The use of machine readable dictionaries in the Pangloss project.* AAAI Spring Symposium on Building Lexicons for Machine Translation. Menlo Park, CA : AAAI Press, 1993. SS-93-02.

11. *Building a Sense-Distinguished Multilingual Lexicon from Monolingual Corpora and Bilingual Lexicons.* **Sammer, Marcus and Soderland, Stephen.** 2007. Proceedings of Machine Translation Summit XI.

12. *Compiling a Massive, Multilingual Dictionary via Probabilistic Inference.* **Mausam, et al.** Suntec, Singapore : s.n., August 2009. Joint Conference of the Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP).

13. **Everitt, Katherine, et al.** *Evaluating Lemmatic Communication.* Seattle, Washington : Turing Center, University of Washington, 2008.

14. **Christensen, Janara, Fader, Anthony and Lin, Thomas.** *Human-Assisted Word Sense Disambiguation.* Seattle, WA, USA : Unpublished Manuscript, 2009. Final Class Report.

15. *A Rose is a Roos is a Ruusu: Querying Translations for Web Image Search.* **Christensen, Janara, Mausam and Etzioni, Oren.** Suntec, Singapore : s.n., August 2009. Joint Conference of the Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP).

16. **Hirsch, E.D.** *Validity in Interpretation.* New Haven, CT, USA : Yale University Press, 1967.

17. **Albright, Eric and Hatton, John.** WeSay, A tool for engaging communities in language documentation. [Online] 2008. [Cited: December 23, 2009.] http://yamiproject.cs.pu.edu.tw/yami/conference/paper/011.pdf.