**SCHOOL OF COMPUTING AND INFORMATICS**

**UNIVERSITY OF NAIROBI**

**MASTERS OF SCIENCE IN COMPUTER SCIENCE**

# A Parallel Corpus Based Translation Using Sentence Similarity

**By**

**RUORO SIMON WACHIRA**

**P58/73542/2009**

**SUPERVISOR**

**Mr. EVANS MIRITI**

**A Project Submitted In Partial Fulfillment Of The Requirements For The Award Of Master In Computer Science At The University Of Nairobi**

## Declaration

This project as presented in this report is my original work and to the best of my knowledge has not been presented in any other institution of higher learning

Signature _____ Date _____

Ruoro Simon Wachira.

P58/73542/2009

This project report has been submitted in partial fulfillment of the requirements for the Award of Master in computer science at the University of Nairobi with my approval as the supervisor.

Signature _____ Date_____

Mr. Evans Miriti.

Lecturer, school of computing and informatics

University of Nairobi

## Acknowledgement

I would like to express my deep gratitude to my supervisor Mr. Evans Miriti for his precious observations, guidance and suggestions and for the great kindness he has shown me during the whole time.

I wish to thank my family, and friends for their great kindness, love and support they have shown me.

I am especially indebted to my colleagues, for there sincere help and encouragement

May God, bless them abundantly.

## Abstract

When large quantities of technical texts are being translated manually, it is very difficult to produce consistent translations of recurrent stretches of text, such as paragraphs, sentences and phrases, making it not possible to reuse old translations stored as translation memories of previous versions of handbooks and thereby minimizing the chances of producing variant translations of the same source sentence that provide users with better understanding on word usage in sentences.

We developed an English-Swahili example-based machine translation (EBMT) system, which exploited a bilingual corpus to find examples that match the input source-language the Translation examples were extracted from a collection of parallel and sentence aligned in English – Swahili for translation. We used the technique of splitting phrase or paragraph into sentences through the use of N-gram. In previous research, many methods used N-gram clues to split sentences. In this project, to supplement N-gram based splitting methods, we introduced another clue using sentence similarity based on edit-distance. In our splitting method, candidate sentence were generated by splitting paragraph based on N-grams, and select the best one by measuring sentence similarity.

We conducted experiments using two EBMT systems, one of which use a word and the other of which use a sentence as a translation unit. Which showed that the system performs slightly better when using sentence similarity in terms of performance a considerable success rate (above 95% at sentence) was encountered in order to construct a database with truthfully correspondent units sentence. The use of words show also showed a good performance of above 65%.

Also the use of classifying text into their domain/topic did show some improvement. Through the use of translation memory (TM) with repository in which the user store previously translation helping to improve translator productivity and consistency, while a TM system functions as an information retrieval system that tries to retrieve one or more suggestions from a TM database that would assist the translator in his/her current translation task or learning how a sentence can be used in different contexts or domains.

## Table of Contents

## ACRONYMS AND ABBREVIATIONS

| | |
|---|---|
| MT | Machine translation |
| RBMT | Rule-based machine translation |
| SMT | Statistical machine translation |
| EBMT | Example-based machine translation |
| WWW | World Wide Web |
| HTML | Hyper Text Mark-Up Language |
| XML | Extended Mark Up Language |
| NLP | Natural Language Processing |
| CAT | computer Aid Translator |
| SQL | Structured query language |

## Table of figures

# Chapter 1: Introduction

## 1.1 Background of the study

The accelerated growth in the size, content and reach of Internet, the diversity of user demographics and the skew in the availability of information across languages, all point to the increasingly critical need for computer aided translation tool. Corpus-based translation systems use existing parallel texts to guide the translation process. One of the main problems when using a corpus-based system for translation is the relatively small quantity of data that the system may have available for the purpose.

Text translation is critical for the acquisition, dissemination, exchange and understanding of knowledge in the global information society And this form the basis of much multilingual research in natural language processing, ranging from developing multilingual lexicons to statistical machine translation systems Maarten (2005).

When large quantities of technical texts are being translated manually, it is very difficult to produce consistent translations of recurrent stretches of text, such as paragraphs, sentences and phrases. This can have many different reasons, for example, several translators work on different sections of the same document simultaneously, the source text is not final and may be changed at a later stage, and it may be too time-consuming or practically impossible to identify recurrent units in the source text manually. Individual translators making up a translation team will also have individual criteria for choosing a certain translation or even choosing from a set of possible translations Magnus Merkel (1993).

One suggested remedy to the problem of consistency in translation is to use tools based on translation memories Magnus Merkel (1996). When using such systems the translators translate the text interactively with a computer tool that stores and retrieves all identical source sentences with their corresponding translations, which guide the translator towards consistent translations. It is also possible to reuse old translations stored as translation memories of previous versions of handbooks and thereby minimizing the chances of producing variant translations of the same source segments. The quality of the translation memories that are being put to use in a translation

project becomes crucial for the quality of the new target text. Translation memories are produced either by using example based translation tool interactively or by aligning a source text with its corresponding translation with text alignment tool.

Parallel text is one of the most valuable resources for development of statistical machine translation systems and other NLP application Budiono et al (2009). However, manual translations are very costly, and the number of known parallel text is limited.

Eneko Agirre et al (2008) indicated that Natural Language Processing (NLP) tasks require a large collection of annually annotated text to train and test supervised machine learning models. While these models have been shown to perform very well when tested on the text collection related to the training data (what we call the source domain), the performance drops considerably when testing on text from other domains (called target domains).

## 1.2 Problem statement

When large quantities of technical texts are being translated manually, it is very difficult to produce consistent translations of recurrent stretches of text, such as paragraphs, sentences and phrases, It is also not possible to reuse old translations stored as translation memories of previous versions of handbooks and thereby minimizing the chances of producing variant translations of the same source sentence.

Unlike this approach, traditional translation and dictionaries are limited and users often cannot find explanations concerning words usages.

## 1.3 Main Objectives,

To developed an experimental English-Swahili example based machine translation (EBMT) system, which exploits a bilingual corpus to find examples that match fragments of the input source language

### 1.3.1 Specific objectives

1. To investigate, to what extent sentences can be extracted from parallel corpus on multiple languages.

2. To provide an array of sentences, and allow the user to select the best equivalent sentence for the source sentence, and see in what circumstances a word would typically be used in practice.
3. To create a library of multilingual sentences to facilitate translation for English-Swahili languages.

## 1.4 Justification

Sentences extracted from parallel corpus in specific domains provide an intuitive mean of grasping the context of a word and has be frequently used to complement conventional word definitions.

However, where a given word has multiple meanings in different contexts, we provide sample sentences in the various contexts in which the word has been used and hence definitions rather than displaying all together.

## 1.5 Scope

The study focused on exploiting parallel repositories in the World Wide Web. To build a multilingual dictionary based on domain specific model.

The system focused on the following areas.

1. Assembling parallel corpus through news feeds.
2. Provision of examples of the usage of words in different contexts and inclusion of translational equivalents from several languages.
3. Domain specific i.e. sports, politics etc.
4. Lexical alignment
5. Building a multilingual database

# Chapter 2: Literature review

Many techniques have been proposed for text and sentences similarity including stemming translation models and query expansion Delphine Bernhard (2006). This section describes several of these techniques that are most related to our work. The task we focus on is a similarity task, in which we compare sentence segments, Translation models, in a monolingual setting, have been used for translation and detecting text reuse.

In a Given process model or fragment finding models or fragments that closely resemble the search model in the repository bring a problem of similarity search. According to Remco et al (2001), the need for similarity search arises in multiple scenarios. For example, when adding a new process model into a repository, similarity search allows one to detect duplication or overlap between the new and the existing process models.

## 2.1: Text similarity

Answering a similarity search query involves determining the degree of similarity between the search model and each model in the repository Remco et al (2001) indicated that similarity can be defined from several perspectives, including the following.

1. Text similarity: based on a comparison of the labels that appear in the process models (task labels, event labels, etc.), using either syntactic or semantic similarity metrics, or a combination of both.
2. Structural similarity: based on the topology of the process models seen as graphs, possibly taking into account text similarity as well.
3. Behavioral similarity: based on the execution semantics of process models.

The approaches previously used for sentence alignment (sentence length, word correspondence and cognate matching) take into account different aspects of similarity between the source and the target language sentences. Anil Kumar Singh and Samar Husain (2007) discussed various aspects of similarity in translated texts that can be used for sentence alignment. They describe a customizable method for combining several approaches that can exploit the aspects of similarity. Their method also includes a novel way of using sentence length for alignment. This involves combining sentence length, word correspondence and cognate matching with some other approaches such as common word count, synonym intersection, and hypernym intersection. The second aspect is making use of language resources such as bilingual dictionary and the WordNet.

Anil et al indicate there are several aspects of the translation process (and correspondingly of similarity) between Source Language and Target Language texts. They believed that under different conditions and for different language pairs, one or more of these may become more suitable for identifying translations, and therefore, also for sentence alignment. Thus, a robust and adaptable sentence alignment tool should exploit as many of these aspects of similarity as is practically possible.

If there are two (written) sentences in two different languages, one of which is a translation of the other, then the similarities between the two may be due to the following factors:

1. Meta-syntactic, one simple measure of which may be the number of verbs (or nouns)
2. Borrowed words, Phonetic and lexical correspondence
3. Syntactic, such as the order or structure of syntactic constituents
4. Meta-semantic, such as the quantity of information, one simple measure of which is the sentence length
5. Semantic, i.e., the common meaning and the world knowledge

Thanh Dao (2005) describes a way of capturing the similarity between two strings (or words). String similarity is a confidence score that reflects the relation between the meanings of two strings, which usually consists of multiple words or acronyms. Currently, in his approach, he was more concerned on the measurement which reflects the relation between the patterns of the two strings, rather than the meaning of the words.

According Thanh Dao (2005) similarity is calculated in three steps:

- Partition each string into a list of tokens.
- Computing the similarity between tokens by using a string edit-distance algorithm (extension feature: semantic similarity measurement using the WordNet library).
- Computing the similarity between two token lists.

According to Matthew R et al (2011), they designed a system called Engkoo that supports a multitude of NLP and Speech technologies such as cross language retrieval, alignment, sentence classification, statistical machine translation, text-to-speech, and phonetic search. The data set that supports this system is primarily built from mining a massive set of bilingual terms and sentences from across the web. Specifically, web pages that contain both Chinese and English are discovered and analyzed for parallelism, extracted and formulated into clear term definitions and sample sentences. This approach allowed them to build the world's largest lexicon linking

both Chinese and English together - at the same time covering the most up-to-date terms as captured by the net. There data set is intelligently merged with licensed data from sources including Microsoft Office and Encarta. Finally, the resulting vast, ranked, high quality composite data set is analyzed by a machine learning based classifier, allowing users to filter down sample sentences by combinable categories.

Nagao (1984) initiated the example-based approach to machine translation with a structural Japanese-to-English translation system. Other influential works include (Sato and Nagao, 1990; Maruyama and Watanabe, 1992; Sumita and Iida, 1995; Nirenburg et al. 1994; Brown, 1999).

The work by Lee (2004) is on improving a statistical Arabic-to-English translation system, based on words as well as phrases, by making the parallel corpus syntactically and morphologically symmetric in a preprocessing stage. This is achieved by segmenting each Arabic word into smaller particles (prefix, stem and suffix), and then omitting some of them in order to make the parallel corpus as symmetric as possible. That method seems to increase evaluation metrics when using a small corpus. Similar conclusions were reached by Sadat and Habash (2006) in their work on improving a statistical Arabic-to-English translation system. There, several morphological reprocessing schemes were applied separately on different size corpora. Philips et al. (2007) present an Arabic-to-English example-based system.

They broaden the way the system performs matching. That system matches words based on their morphological information, so as to obtain more relevant chunks that could not otherwise be found. It showed some improvement over state-of-the-art example-based Arabic-to- English translation systems. This matching approach also resulted in additional irrelevant matched fragments, which had to be removed in later stages.

There are a number of works on automatic thesaurus creation. Some of them use parallel corpora for finding semantically-related source-language words based on their translations. Lin and Pantel (2001) extracted paraphrases from a monolingual corpus by measuring the similarity of dependency relationship. They use a syntactical parser to parse every sentence in their corpus and measure the similarity between paths in the dependency parses using mutual information.

Paths with high mutual information were defined as paraphrases. Glickman and Dagan (2003) described an algorithm for finding synonymous verbs in a monolingual corpus. This was also done using a syntax parser for building a vector containing the subject, object and other arguments for every verb they find in their corpus. Later, they use these vectors to find

similarities between verbs. Overall this technique showed competitive results with the one introduced by Lin and Pantel (2001). Nonetheless, since both techniques may perform differently on a given case, they suggested that the methods should be combined to obtain better results. One interesting work is Dyvik (2006), which uses an English-Norwegian parallel corpus for building a lattice of semantically-related English and Norwegian words. It then discovers relations such as synonyms and hyponyms. Another related work van der Plas and Tiedemann, (2006) uses a multilingual sentence-aligned parallel corpus for extraction of synonyms, antonyms and hyponyms for Dutch.

According to Palakorn et al (2008), the ability to accurately judge the similarity between natural language sentences is critical to the performance of several applications such as text mining, question answering, and text summarization. According to their study given two sentences, an effective similarity measure should be able to determine whether the sentences are semantically equivalent or not, taking into account the variability of natural language expression. That is, the correct similarity judgment should be made even if the sentences do not share similar surface form. In this work, they evaluate fourteen existing text similarity measures which have been used to calculate similarity score between sentences in many text applications.

One extension of word co-occurrence methods leads to the pattern matching methods which are commonly used in text mining and conversational agents Corley and Mihalcea (2005). This technique relies on the assumption that more similar documents have more words in common. But it is not always the case that texts with similar meaning necessarily share many words. Again, the sentence representation is not very efficient as the vector dimension is very large compared to the number of words in a short text or sentence, thus, the resulting vectors would have many null components.

While Yee Seng Chan and Hwee Tou Ng,(2009), presented a Maximum Similarity Metric for Machine Translation Evaluation (MAXSIM), a new automatic machine translation (MT) evaluation metric that computes a similarity score between corresponding items across a sentence pair, and uses a bipartite graph to obtain an optimal matching between item pairs. There general framework allowed them to use arbitrary similarity functions between items, and to incorporate different information for comparison. They evaluated for correlation with human judgments', MAXSIM achieved superior results when compared to current automatic MT evaluation metrics.

Al o Gliozzo and Carlo Strapparava (2006) attempted to merge the subspaces associated to each language to exploit the information provided by external knowledge sources, in bilingual dictionaries, by collapsing all the rows representing translation pairs. In this setting, the similarity among texts in different languages could be estimated by exploiting the classical Vector Space Model (VSM) just described. However, their main disadvantage of this approach was to estimate inter-lingual text similarity as it strongly relies on the availability of a multilingual lexical resource. For languages with scarce resources a bilingual dictionary could be not easily available. Secondly, an important requirement of such a resource is its coverage, another problem they noted was that ambiguous terms could be translated in different ways, leading them to collapse together rows describing terms with very different meanings.

Alo et al (2006) in their study indicated that the similarity among two texts in the VSM is estimated by computing the cosine of their vectors in the VSM. They concluded that, such a model cannot be adopted in the multilingual settings, because the VSMs of different languages are mainly disjoint, and the similarity between two texts in different languages would always turn out to be zero. Another work by Masao Utiyama and Hitoshi Isahara (2003) proposed method, which uses similarities in sentences aligned by dynamic programming matching, as a reliable measure for article alignment.

## 2.2: Similarity measure

Sentence similarity measures play an increasingly important role in text-related research and applications in areas such as text mining, Web page retrieval, and dialogue systems. Existing methods for computing sentence similarity have been adopted from approaches used for long text documents. These methods process sentences in a very high-dimensional space and are consequently inefficient, require human input, and are not adaptable to some application domains. This section focuses directly on reviewing different work done on similarity measures between sentences. Reviewing present algorithms that take account of semantic information and word order information implied in the sentences. The semantic similarity of two sentences is calculated using information from a structured lexical database and from corpus statistics.

Determining the similarity between sentences is one of the crucial tasks which have a wide impact in many text applications. In information retrieval, similarity measure is used to match score between a query sentence and equivalent sentence in second language in a corpus this requires sentence matching between primary sentence and target sentence

In previous a study by Rejwanul et al (2010) indicates in their study that phrase pairs could be extracted from many training sentence pairs, by calculating the similarity score between the source test sentence and each of the training sentences separately, and then take the average of the scores. Finally, they normalize these similarity scores to convert them into probabilities Thus, deriver a log-linear feature. They explored various sentence similarity features by measuring similarity between a source sentence to be translated with the source-side of the bilingual training sentences and integrate them directly into the phrase based model. They performed experiments on an English-to-Chinese translation task by applying sentence-similarity features both individually and collaboratively with super tag-based features.

Li et al. (2006) propose another hybrid method that derives text similarity from semantic and syntactic information contained in the compared texts. Their proposed method dynamically forms a joint word set only using all the distinct words in the pairs of sentences. For each sentence, a raw semantic vector is derived with the assistance of the WordNet lexical database Miller et al. (1993). A word order vector is formed for each sentence, again using information from the lexical database. Since each word in a sentence contributes differently to the meaning of the whole sentence, the significance of a word is weighted by using information content derived from a corpus. By combining the raw semantic vector with information content from the corpus, a semantic vector is obtained for each of the two sentences. Semantic similarity is computed based on the two semantic vectors. An order similarity is calculated using the two order vectors. Finally, the sentence similarity is derived by combining semantic similarity and order similarity. These two hybrid measures Li et al. 2006; Mihalcea et al. 2006 do not take into account the string similarity, which plays an important role in some cases. We discuss why string similarity is important in next section.

Feature-based methods try to represent a sentence using a set of pre-defined features. Similarity between two texts is obtained through a trained classifier. But finding effective features and obtaining values for these features from sentences make this category of methods more impractical.

Aminul Islam and Diana Inkpen (2008), propose method to determine the similarity of two texts from semantic and syntactic information (in terms of common-word order) that they contain. They consider three similarity functions in order to derive a more generalized text similarity

method. First, string similarity and semantic word similarity were calculated and then they use an optional common-word order similarity function to incorporate syntactic information in their method, Finally, the text similarity is derived by combining string similarity, semantic similarity and common-word order similarity with normalization

In their paper Jiannan Wang et al (2011), studied the problem of string similarity join. The proposed a new similarity function by combing token-based similarity and character-based similarity. They proved that existing similarities are special cases of fuzzy token similarity. Jiannan Wang (2011), proposed a signature-based framework to address the similarity join using fuzzy-token similarity. They also proposed token-sensitive signature scheme, which is superior to the state-of-the-art signature schemes. They extended existing signature schemes for edit distance to support edit similarity. They also devised a partition-based token signature scheme.

### 2.3.1 Similarity Algorithms

In this section we review different ways of quantifying how similar two strings (e.g., words) are to one another by counting the minimum number of operations required to transform one string into the other. Through use of algorithms to determine candidate sentence or words for a given lexicon database by selecting words from a dictionary that have a similarity to the word/sentence in question.

#### *2.3.1.1: Levenshtein*

This algorithm measures the difference between two sequences/strings, and is based around the number of changes required to make one string equal to the other. It is aimed at short strings, it usage is spell checkers, optical character recognition

Thanh Dao (2005) method uses an edit-distance string matching Levenshtein algorithm. He argued that the string edit distance is the total cost of transforming one string into another using a set of edit rules, each of which has an associated cost. He argued that Levenshtein distance is obtained by finding the cheapest way to transform one string into another. Thanh Dao (2005) indicated that transformations are the one-step operations of (single-phone) insertion, deletion and substitution. In the simplest version substitutions cost about two units except when the source and target are identical, in which case the cost is zero. Insertions and deletions costs half that of substitutions.

In his paper Fernando (2009) explored Levenshtein distance and explained that Levenshtein algorithm allows insertions, deletions and substitutions. Similarity is often measured using Levenshtein distance, which refers to the algorithm written by Russian scientist Vladimir Levenshtein in 1965. With Levenshtein distance, you can count the number of insertions, deletions, and substitutions required to turn one phrase into another. Where the difference is usually measured against the source phrase, so a CAT tool using this method would indicate that the similarity between the two words.

According to him Levenshtein algorithm is very practical, but not practical enough. For example, a human translator can tell that the following two sentences are similar in meaning:

4. "Ice cream: chocolate and vanilla"
5. "Ice cream: vanilla and chocolate"

However, a program computing the number of keystrokes required to change one phrase into the other would say that those sentences are quite different, and may not offer translations from the database. Rearranging the words requires too many keystrokes.

With languages like English or French, it is easy to separate the words that compose a sentence, making it possible to take word order into account when measuring similarity.

#### *2.3.1.2: Hamming*

According to Fernando 2009, Hamming distance allows only substitutions, so the Hamming distance between two strings of the same length is the number of positions for which the corresponding symbols are different. If the Hamming or edit distance are used to calculates the distance between two strings, the string have to be of equal length.as it measures the minimum number of substitutions for the two strings to be equal.

#### *2.3.1.3: Jaro Winkler*

This algorithm is purposely designed for record linkage; it was designed for linking short strings. It calculates a normalized score on the similarity between two strings; the calculation is based on the number of matching characters held within the string and the number of transpositions.

In a paper by Andriy et al (2008) compared Jaro Winkler with other string similarity metrics (edit distance and Levenshtein) and found that it out performs others. Therefore in their test they us edit as a representative of string similarity matching methods. In order to cover the cases when the tokens in two multi-word string labels have different formats they used Jaro Winkler

algorithm, when both compared values are tokenized, each pair of tokens is compared using the standard Jaro Winkler measure and the maximal total score is selected. they assumed that the algorithms did not have any domain specific knowledge, so for each individual only its immediate data-type properties were considered.

### *2.3.14: Markov Chain*

The Markov Chain model calculates the probability of the next letter occurring in a sequence based on the previous n characters. They are used in a multitude of areas, including data compression, entropy encoding, and pattern recognition.

Kevyn et al (2007) described a probabilistic model of text semantic similarity that uses Markov chains on a graph of term relations to perform a kind of semantic smoothing. This model incorporated both corpus-based and knowledge-based resources to compute text semantic similarity. They measured the effectiveness of both our method and LSA compared to cosine and random baselines, using a new corpus of human judgments on definition responses from a language learning experiment.

## 2.4: Translation memory

Translation memory (TM) is a language technology that enables the translation of segments (sentences, paragraphs, or phrases) of documents by searching for similar segments in a database and suggesting matches that are found in the database Rodolfo Raya (2004).

TM is a fundamental part of modern computer aided translation (CAT) tools. It has become so common in the translation industry that the term "translation memory tool" is often used in place of "computer aided translation tool." However, these terms should not be used interchangeably, as CAT technologies also include machine translation, a computer technology based on linguistic rules and the use of bilingual dictionaries.

A TM system remembers translations that have been typed by a human translator. When the translator needs to work on a similar text, the system offers the previously saved version. This can save a lot of time when a translator works with repetitive texts, such as technical manuals, and can also help to achieve terminological consistency.

Karl-Johan Lönnroth (2005) argued that TM systems are basically search engines that specialize in performing two kinds of searches:

- **Exact searches:** Only entries that match the searched text are retrieved from the data repository.

**Fuzzy searches:** Entries that are similar to the searched text up to certain extent —specified in the query— are retrieved from the data repository.

The level of similarity in fuzzy searches is called **match quality**. To him definition varies with each TM system implementation.

Tanaka et al (2000), indicates that the key to providing better matches lies in the ability of the TM engine to break every segment into very small pieces and store them as a large collection of fragments in a database. The drawback to this is that the data storage requires that you spend a lot of time and effort doing proper indexing. Nevertheless, searching for fragments is faster than full text search combined with similarity calculations. They noted, during the translation process, it is often necessary to transfer translation memories along with the documents being translated. Because the various participants in the process might have different TM systems, a standard for TM data exchange was created.

According to Magnus Merkel (2003) being consistent in technical translations is difficult. Using translation memory software could be one remedy, but the effectiveness of these tools rests on the assumption that most repeated source sentences correspond similar to repeated target sentences. With the help of a discrepancy tool to identify inconsistencies between source and target texts, data from translations made manually and with the aid of translation memories are presented. Apart from being used as a verification tool at the post editing phase of a translation project, the discrepancy tool will measure the relative efficiency of the use of translation memory tools, as well as identify possible shortcomings with the source text.

## 2.5: Summary

According to our research there are a lot algorithms available for text similarity after from our analysis and review we chose to use the edit distance in order to compare the input sentence with different examples in the translation memory for EBMT but according to Cristina et al (2001) two problems appear when using this method.

1. It measures differences between strings and not words, which means that words which are semantically very close (for example inflected forms) could be measured by the edit distance quite different. The problem is more complicated when synonyms are used as they are judged completely different by the edit distance.
2. Also when the translation memory is built from a parallel corpus, the constituents are quite big sentences. But also previous research has shown that levenshtein edit

distance algorithm. Can be used to solve sentence difference by computing the difference between two given sentence. Given a language of valid (or correct) words, we can correct a given word to some word of the language that is the least different to the given word.

Friedel et al (2014) provided insight into areas where the recall of translation memory systems can be improved. Specifically, source texts with an omission, and semantically very similar source texts are some of the more frequent cases with useful target text suggestions that are not selected with the baseline approach of simple edit distance between the source texts.

The edit distance between two strings source and target sentence is defined as the minimum number of operations performed in order to obtain s2 from s1 (Levenshtein algorithm) Stavros (2007). The measure gives an indication for the closeness of the two strings. Which will helped us to achieve our objective of providing sample similar sentence that correspond to the main sentence.

# Chapter 3: Methodology

## 3.1: System Analysis

### 3.1.1: Overview

Our aim was to build an easy to use translator of where user will contribute spontaneously in building lexical sentence in the languages they know. We expect users to send monolingual search requests in any language supported by our system to get multilingual answers. Through the use of our search engine user will extract their requests and will be able to add the new searches to the dictionary spontaneously. We chose to use Iterative design as it is based on a cyclic process of prototyping, testing, analyzing, and refining a product or process Nielsen (1993).

## 3.1 Functionality description

The system provides the end user with the following functionalities:

1. Content management interface this enables the user to organize, modify content, delete an as well maintain files from news websites where primary sentence (English) and secondary sentence (Kiswahili) are extracted.
2. Machine translation for English and Swahili

The system allows the user to query similar sentence in one natural language (primary language) for sentence in another (target language).

.

## 3.2 System design

### 3.2.1 Context diagram

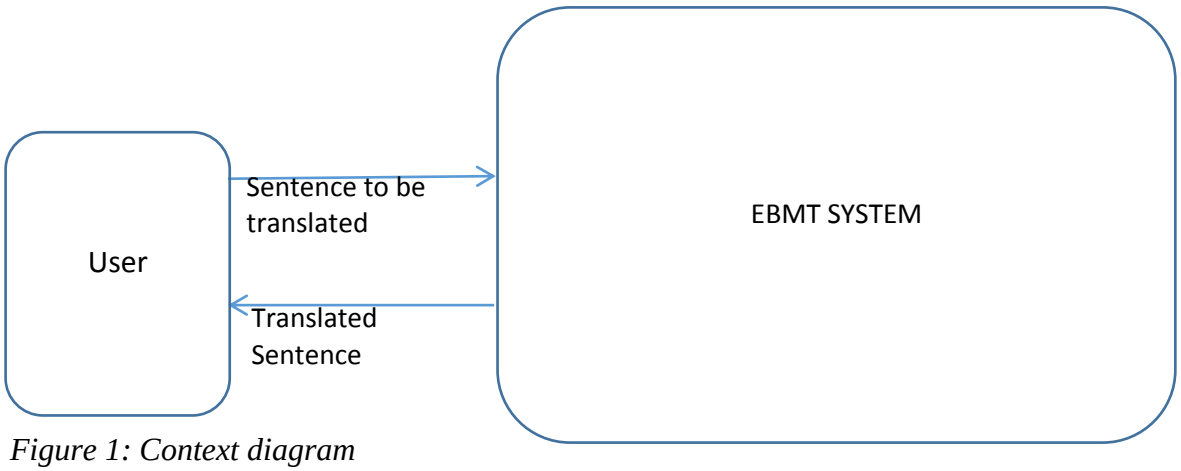


*Figure 1: Context diagram*

### *3.2.2: Data flow diagram*

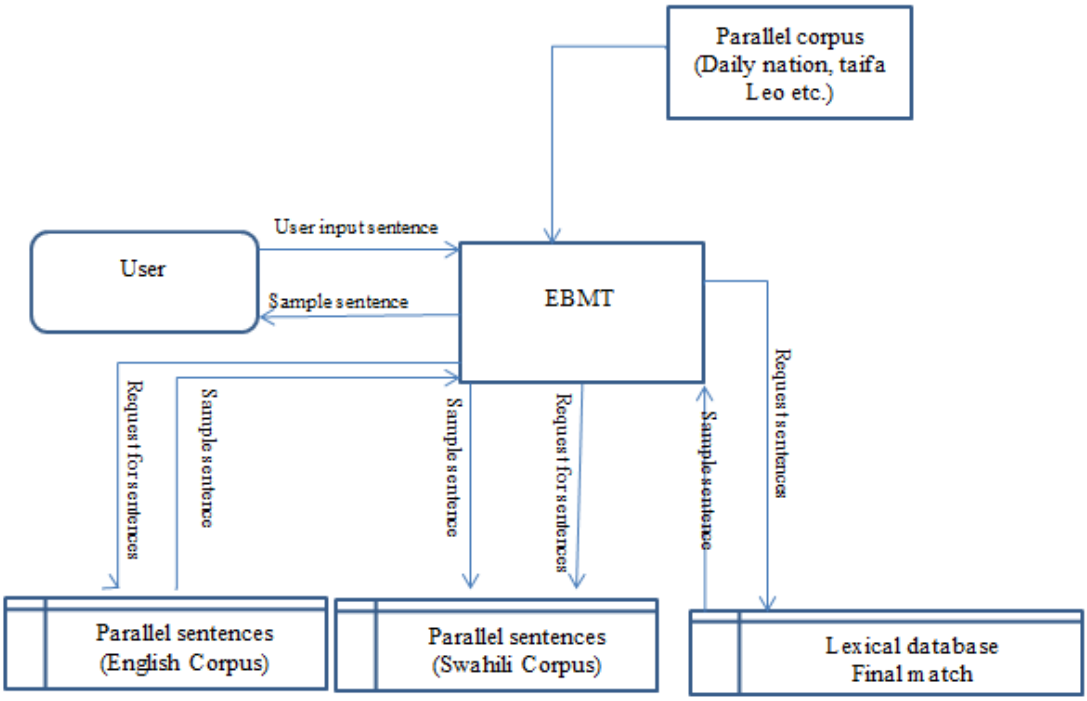


***Figure 2:*** *Data flow diagram*

### 3.2.3 Entity relationship diagram

Figure three below show design of the lexical database.

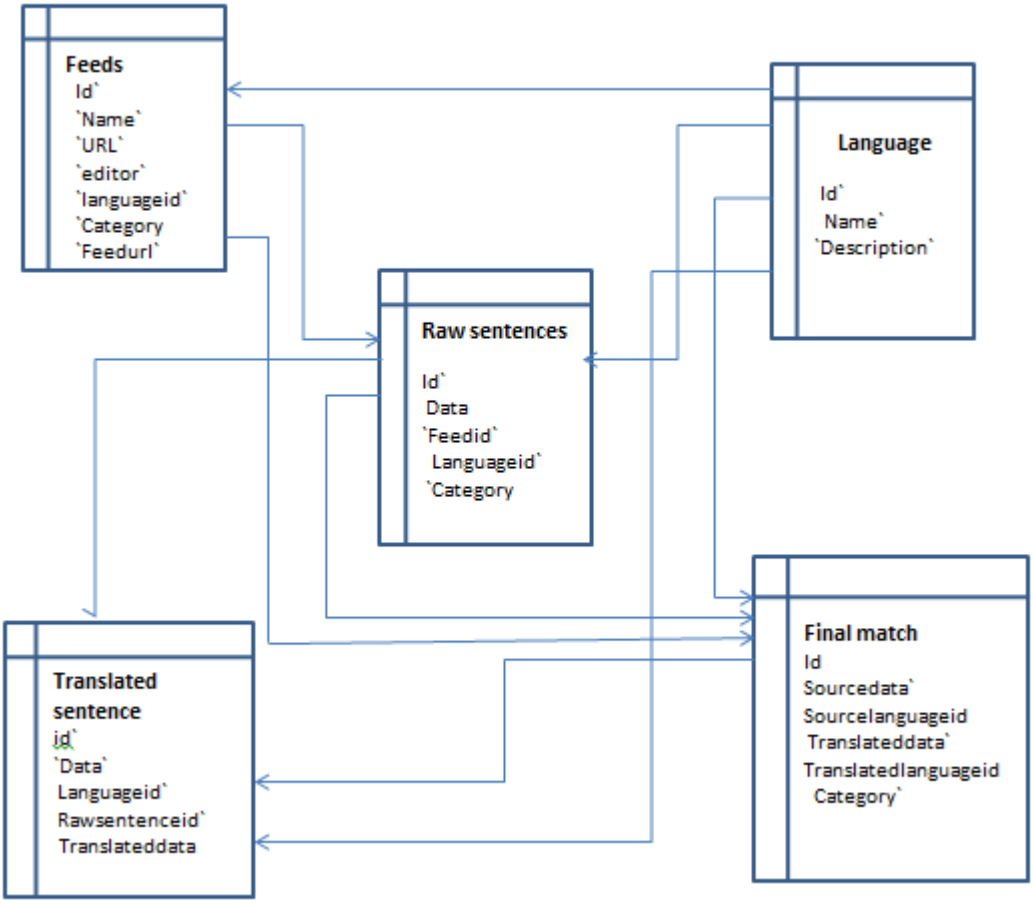


*Figure 3: Entity relationship diagram*

Lexical database

This provides organized collection of matched English-Swahili sentences organized to model aspects of reality in a way that supports processes of translation accurate and more consistent way. The requirement is to be open to the user and to describe much more attributes of sentence as required for the EBMT and also extend the same database for other pairs of languages as well

We define our queries to the data using SQL (Standard Query Language), so the exact DMS is not essential (we use MySql). The usage of DMS make the lexicon easy to modify: to add new

attributes, delete them, or modify the names or types and etc. It is possible to extend the same database to other human languages as well.

We considered the domain possibility for the sentence, thus user can choose the priorities of finding words or sentences in certain domain. This way user can prioritize the word translation according to his domain of interest.

## 3.3 Multilingual Translation Structure

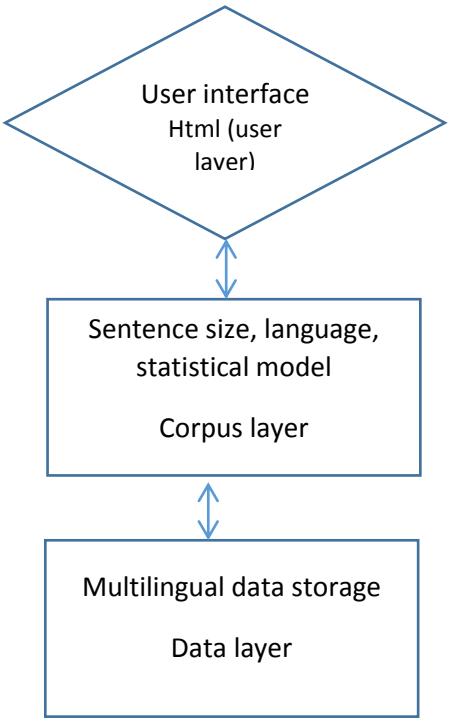


***Figure 4: Multilingual Translation Structure***

**User layer**

User layer interact with users by a simple set of HTML pages and web forms that provide sentences retrieved at the corpus layer.

**Corpus layer**

At this layer corpus are preprocessed in order to get additional linguistic information about the source language, e.g. about frequencies of collocations while tokeniser takes a SL sentence as input. One of its tasks is the separation of words and punctuation adding tags to words and

sentences. And use statistical models n-gram and levenshtein algorithm to find similar sentences in Swahili to the source sentence in English.

We used news feed from the manually created list, we extract paragraphs from pages. Once a paragraph is extracted, they are split into sentences and then stored. We then determine sentence size, language and its domain topic. We also keep trace of web pages where a sentence has been extracted from, these provide the candidate sentences.

We then take the candidate sentence in language identified and then find corresponding sentences in languages supported this is achieved by first aligning sentences, and then using Statistical Machine translation to connect words to corresponding sentences. And the result produced a multilingual dictionary.

**Data layer**

At the data layer, multilingual dictionary entries were indexed into a database, which give dictionary lock-up facilities, the structure can be changed dynamically to include any kind of information needed later. Its simplicity make it easy for users to learn, their searches are automatically transfer into an XML document and indexed into the dictionary.

This provides a variety of Web results and sentences to help users get context for a word or phrase. We also suggest including text-to-speech functionality in future, so users can hear a word or phrase read out loud in a sentence.

When a user inputs a sentence in our the base language(English) for which he wants to translated in a specific domain, the system sends this sentence query to our database and obtains search results in sample sentences. The usages are extracted by performing by matching the input sentence with the sentence extracted and then provide feedback to the user. Unlike existing tools, the translator is multilingual, so that usages are obtained even in language for which there are no well-established analytical methods.

Our system uses SMT for its data-driven approaches, and which is much faster in development, as it do not rely on hand-crafted rules. Instead, these approaches rely on large parallel corpora, which therefore are the bottleneck in developing new language pairs or new technical domains, as they are often unavailable, or not large enough.

## 3.4 Data collection

The translation examples in our system were extracted from a collection of parallel, sentence-aligned, English-Swahili html documents, taken from a news-related corpus published by different news agency i.e. daily nation ,taifa leo etc. . All the Swahili translation examples were morphologically analyzed, and each translation example was aligned on the data level.

Text parallelization and parallel corpora is important for NLP, language studies, and dictionary creation. A tremendous amount of information was required and was collected from WWW through RSS news feeds sources. This data was used to meet the requirements of the system.

John Fry (2005) exploited the recent trends in the delivery of news over the World Wide Web (WWW).taking advantage of the growing practice of multi-national news organizations that publish content in multiple languages across news sites. We use domain categories that define the group of our aggregated RSS News Feeds.

Given a new input sentence, the system begins by searching the corpus for translation examples for which the Swahili (primarily translated sentence matches frag the input sentence). In the implementation we are describing, the system is restricted to parts of the input sentence so that a matched sentence must be a combination from one or more complete adjacent sentence of the input sentence. The base-sentence are initially extracted using the Lucerne tool.

The same fragment can be found in more than one translation example. Therefore sentence are matched word by word, to deal with data sparseness, we generalize the relatively small corpus by matching words on text, morphological, and sentence levels, with each level assigned a different score.

Under the Text/sentence level which falls under our data layer in our multilingual structure we measured the difference between two sequences/strings or word Where the difference is usually measured against the source phrase, so a CAT tool using this method would indicate that the similarity between the two words in the sentences produce sample sentences in our translated language Fernando (2009).

Finding a synonym for a given sentence in one language is not a simple task, considering that input sentences are not given with word senses. Matching input words based on synonymy without knowing their true senses is error-prone, because one might match two synonym words based on a specific sense that is not the one used by the author Kfir Bar et al (2007). The way to

handle this issue would be to use a jdom tool for swahili to uncover the intended sense of each input sentence word.

Although there has been some research in this area, we could not find any available tool that produces reasonable results. Even were we to find one, it would probably use English WordNet senses, since Swahili WordNet is not ready yet.

In this work, we decided to experiment with a different route where we classified each input sentence by topic, as well as all the corpus translation examples. For each translation example, we consider synonyms only if it's topic-set intersects with that of the input sentence.

Instead, users can take advantage the system capabilities to translate by providing sample sentences and also providing a word usage for Swahili words in sentences. Our system provides search and translation capabilities, but also diverse results to queries in terms of topics or domains. Of course, the focus here is to offer context, rather than simple translation. We allow users explore language by discovering new sentences using wildcard by simply input your sentence.

## 3.5 System Implementation

The developed system at large constitutes a multilingual environment for Swahili and English languages. It can be viewed as an on-line multilingual lexicon, able to manage, store, retrieve and depict two language elements representations and multimedia representation that's include text, and sentences . It can also hold user specific information concerning vocabularies, plus multiple modifiable dictionaries per learner. The system also provides omnidirectional "word-by-word" translation between the elements of the various defined languages, after specifying the source and target languages and desired concept for translation. The system is open and expandable by means of user profiles, languages, and language elements and representations. Since the system uses multimedia elements (as one can realize considering the variety of representational media) can fall into the multimedia application category. Interactive multimedia have earned a prominent status among software developers due to their variety and flexibility and the ease of incorporating different learning modes. However since there's no guaranteed way of applying such a technology effectively, constant evaluation is imperative. In this respect, and

in order to develop the software described herein, there was an extensive use of the generic methodology and instrument for the multimodal evaluation of interactive multimedia.

### 3.5.1 Technology

The system was developed as a 32-bit application to run under window 7/xp. The architecture that was chosen for the application was the Server Side technology. We also incorporated java and xml technology, plus mixture of JavaScript code and Sql,

### 3.5.2 Development tools

The project required a diverse resources; this was categorized as the following.

Hardware:

- Laptop: specification Intel core2Duo CPU @ 2.2 GHZ, 4 GB RAM, 500G Hard disk capacity running of windows 7 professional 32-bit as the developer machine and system host.

Software:

- NetBeans 7.4 is an open-source integrated development environment (IDE) for developing primarily with Java, but also with other languages, in particular PHP, C/C++, and HTML5.It is also an application platform framework for Java desktop applications
- Apache 2.0 - an open-source web server for hosting the web user interface.

Databases:

- RDBMS – open source version of MySQL 5.1.
- eXist-db is an open source database management system built using XML technology. It stores XML data according to the XML data model and features efficient, index-based XQuery processing.

# Chapter 4: Result, Testing and discussion

## 4.1: Testing

Testing of the system was based on following aspects;

- Unit testing: We carried out unit test to catch errors in software, and fix them at the unit level.
- Integration testing: these series of tests was executed to expose errors that were keeping the application from performing its essential functions. The goal was to determine, if the core functionality can be executed without error, uncovering errors associated with the interaction of subsystem classes.
- Validation testing: Was done to provide assurance that the software meets the functional requirements established for this iteration.
- System testing this was to provide verification that all of the disparate, constituent objects of the application mesh properly and that overall system function and performance were achieved

The accuracy of the entire system was measured against the test (real-world) aggregated in our multilingual database. The bench mark being:

- Success: having an acceptable array of similar sentence
- Failure: having failed to produced array of relevant sentences

## 4.2: Result

The data collected was analyzed basing on the objectives of the study. This was done qualitatively by giving meaning to findings to be able to make effective recommendations. We cross checked Data for consistency basing on the following concept

1. Data reliability, availability and integrity
2. Checking whether or NOT a blank value or default value can be retrieved from the database.
3. Validating each value if it is successfully saved to the database.
4. Ensuring the data compatibility against old hardware or old versions of operating systems.
5. Verifying the data in data tables can be modified and deleted and interpreted to provide requirements for designing a multilingual dictionary.

Test 1 News feeds

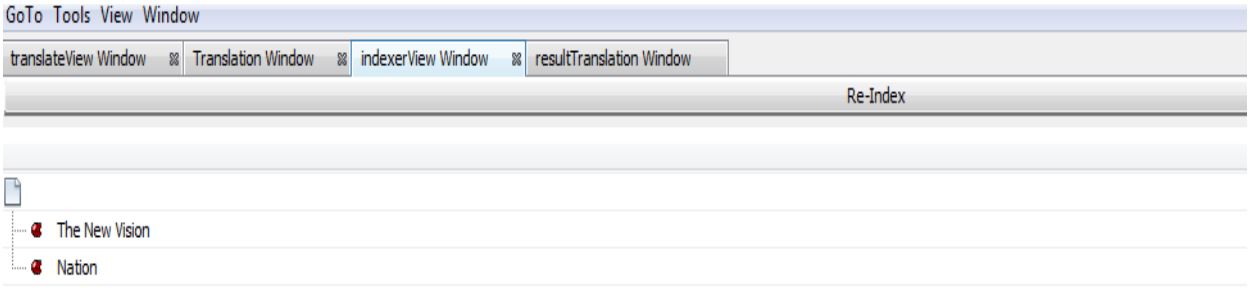


*Figure 5: News feeds*

This test was done to ensure that the news feed were available this was done by click on the index view window which displays the list of available feeds by their publishing media houses.

Test 2: Tagging source sentence to target sentence

After sentence are extracted and split we index the raw sentence (source) to translated sentence (target sentence) by clicking on the Re- indexed button this is to align sentence to their equivalent in the database

Test 3 sample sentences

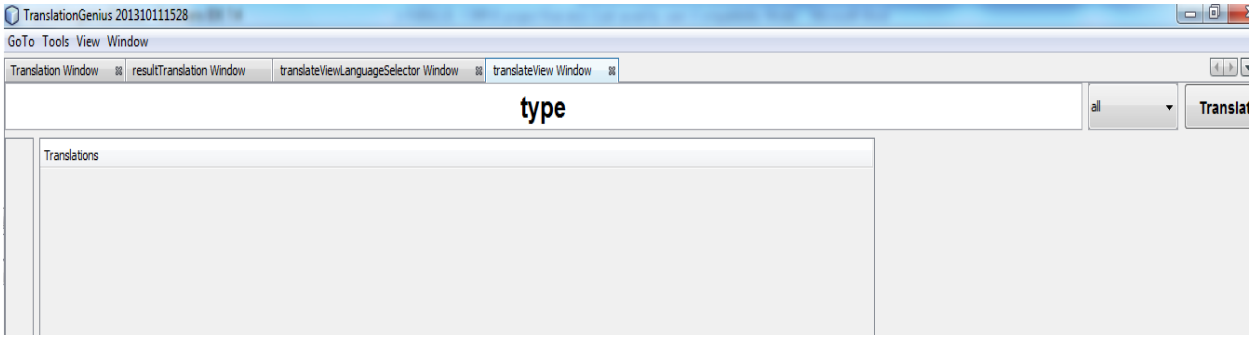


*Figure 6: dialogue to input sentence*

To translate a sentence type a new input sentence the textbox select the category then click on the translate button the system begins by searching the corpus for translation examples for which the Swahili version matches fragments of the input sentence will be displayed as Figure 5 shows.

Test 4 sample sentence generated for an input sentence

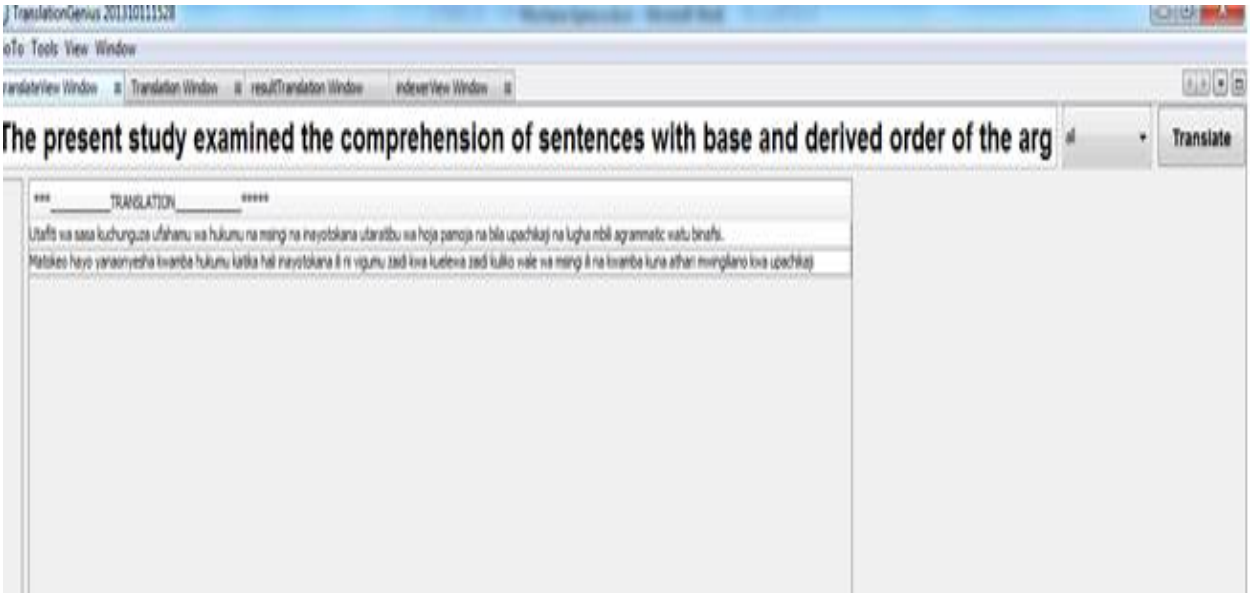


*Figure 7: sample sentence generated for an input sentence*

When the candidate sentences in target language are identified this is by using Statistical Machine translation to connect words to corresponding sentences. And the result produced and displayed for the user the select the best sentence

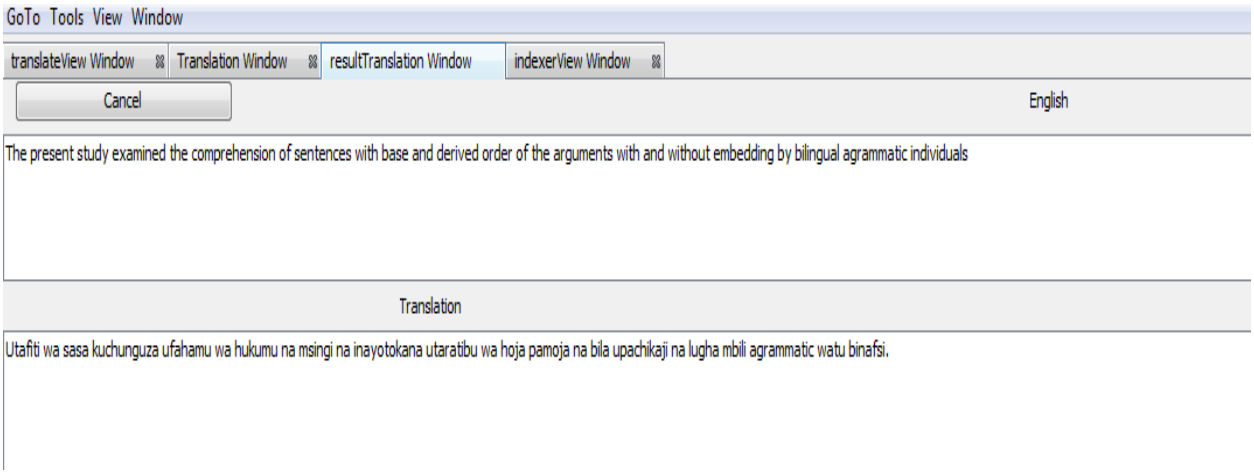


*Figure 8: final translation*

After selecting the best sentence the system provides a chance to improve on the translation as some of the Swahili words are not yet translated

List of scrapped raw sentences in the databases

select * from rawsentence...

Page Size: 20 Total Rows: 145 Page: 1 of 8 Matching Rows:

| # | id | data | feedid | languageid | category | misc |
|---|---|---|---|---|---|---|
| 1 | 144 | A major security operation is currently ongo... | 2 | 1 | news | <NULL> |
| 2 | 145 | Former United States President Bill Clinton ... | 2 | 1 | news | <NULL> |
| 3 | 146 | Major airlines blocked flights to Israel on W... | 2 | 1 | news | <NULL> |
| 4 | 147 | A few months ago, when Emily Heaton said ... | 2 | 1 | news | <NULL> |
| 5 | 148 | The ICC stated that Mr Justice Hans-Peter ... | 2 | 1 | news | <NULL> |
| 6 | 149 | A way of grading all public and private univ... | 2 | 1 | news | <NULL> |
| 7 | 150 | More than 50 buffaloes will be moved from ... | 2 | 1 | news | <NULL> |
| 8 | 151 | Police in Mombasa County were Tuesday on... | 2 | 1 | news | <NULL> |
| 9 | 152 | | 2 | 1 | news | <NULL> |
| 10 | 153 | President Kenyatta to pick eight people fro... | 2 | 1 | news | <NULL> |
| 11 | 154 | Business, religious and civil society leaders ... | 2 | 1 | news | <NULL> |
| 12 | 155 | Dr Swazuri, alleges Mrs Ngilu restricted the ... | 2 | 1 | news | <NULL> |
| 13 | 156 | An oversight agency is monitoring the secur... | 2 | 1 | news | <NULL> |
| 14 | 157 | Matthew Durham charged with sexual abus... | 2 | 1 | news | <NULL> |

*Figure 9: dataset for extracted source sentence (English)*

List of translated sentences

select * from finalmatch

Page Size: 20 Total Rows: 25 Page: 1 of 2 Matching Rows:

| # | id | sourcedata | sourcelanguageid | translateddata | translatedlanguageid | category |
|---|---|---|---|---|---|---|
| 1 | 1 | the world is ours | 1 | dunia ni yetu | 2 | news |
| 2 | 2 | people should pray | 1 | watu wanafaa kuomba | 2 | news |
| 3 | 3 | God loves us all | 1 | Mungu anatupenda sote | 2 | news |
| 4 | 4 | we are in town | 1 | tuko mjini | 2 | news |
| 5 | 5 | Today I built the nation | 1 | Leo nilijenga taifa | 2 | all |
| 6 | 6 | I think I love this music | 1 | Nafikiri ninapenda hii musiki | 2 | all |
| 7 | 7 | try running faster | 1 | jaribu kwenda kasi | 2 | all |
| 8 | 8 | Very well | 1 | Mzuri sana | 2 | hospitality |
| 9 | 9 | They were the best of times | 1 | Zilikuwepo nyakati nzuri | 2 | al |
| 10 | 10 | they were the worst of times | 1 | Ilikuwa ni wakati mbaya | 2 | all |
| 11 | 11 | the most famous opening sentence in Englis... | 1 | kifungua sentensi maarafu sana katika fasih... | 2 | all |
| 12 | 12 | My name is | 1 | Jina langu ni | 2 | education |
| 13 | 13 | What do you call yourself | 1 | Unaitwa nani | 2 | all |
| 14 | 14 | To live long is to see much | 1 | Kuishi kwingi ni kuona mengi | 2 | news |
| 15 | 15 | My sympathy | 1 | pole | 2 | all |
| 16 | 16 | My sympathy | 1 | samahani | 2 | all |

*Figure 10: data set of final matched sentences*

List of matched sentence in the databases

select * from feeds

Page Size: 20 Total Rows: 10 Page: 1 of 1 Matching Rows:

| # | id | name | url | editor | languageid | category | feedurl |
|---|---|---|---|---|---|---|---|
| 1 | 10 | allafrica | http://allafrica.com | allafrica | 1 | ict | http://allafrica.com/ict/ |
| 2 | 2 | Nation | http://www.nation.co.ke/ | Nation | 1 | news | http://www.nation.co.ke/-/1148/1148/-/vie. |
| 3 | 3 | standard | http://standardmedia.co.ke/ | standard | 1 | sports | http://www.standardmedia.co.ke/rss/sport.. |
| 4 | 4 | standard | http://standardmedia.co.ke/ | standard | 1 | Business | http://www.standardmedia.co.ke/rss/busin.. |
| 5 | 5 | standard | http://standardmedia.co.ke/ | standard | 1 | politics | http://www.standardmedia.co.ke/rss/politic. |
| 6 | 6 | standard | http://standardmedia.co.ke/ | standard | 1 | news | http://www.standardmedia.co.ke/rss/world. |
| 7 | 7 | standard | http://standardmedia.co.ke/ | standard | 1 | entertainment | http://www.standardmedia.co.ke/rss/enter. |
| 8 | 1 | The New Vision | http://www.newvision.co.ug/ | Vision Group | 1 | news | http://www.newvision.co.ug/rss/fp.rss |
| 9 | 8 | the people | http://english.peopledaily.com | people | 1 | news | http://english.peopledaily.com.cn/ |
| 10 | 9 | the people | http://english.peopledaily.com | people | 1 | business | http://english.peopledaily.com.cn/business/. |

*Figure 11: data set of news feeds*

## 4.3: Discussion

For the purposes of this project, we have developed an experimental English-to-Swahili example - based machine translation (EBMT) system, which exploits a bilingual corpus to find examples that match the input source-language the Translation examples were extracted from a collection of parallel and sentence aligned in English – Swahili we conducted our experiment two bilingual corpora both containing translations examples of about 3000 sentence the system was tested in all aspect and also the effect of the topic classifier (i.e. sport ,politics) feature on translation was tested and examined. In the following aspect

1. The comprehensiveness of sentence retrieved from multiple resources, conversion to a desired format and integration to the multilingual database.
2. The accuracy of extracted sentence considering similarity measure
3. The presentation of final match sentence in Swahili and English to the users in multiple categories /topics

Our results are based on a data set test of 200 sentences from our observation; the system performs slightly better when using sentence similarity and as expected the system was able to Provide more translation examples that might match the input sentence with the goal of a qualitative evaluation of the effect of including the sentence matching and topics . Exact-text matches and cardinal matches receive full weight (90%). in terms of performance a considerable success rate (above 95% at sentence) was encountered in order to construct a database with

truthfully correspondent units sentence. We are also made it desirable that the alignment method to be language-independent.

Also the use of classifying text into their domain/topic did show some improvement. We can also conclude that test in which we could not find similar sentence did not help improve translations significantly

Our expectations were that exact sentence is more likely to be found in news articles and documents dealing with similar subject matters. We automatically evaluated the results and realized that there was only a slight insignificant improvement in the final results. We decided to examine the results manually.

Overall, we found 193 input sentences for which at least one of the words matched on a sentence. Out of these input sentences, 57 are covering parts in the input sentence that are not covered by other sentence of at least the same size. That means they might help to better cover the input sentence in the matching step, however our current recombination algorithm was not able to capture that. We further looked at the extracted translation for the 193 sentences and found that only 100 were actually translated correctly. All the other sentences received wrong translations by the system. From a first look, in most cases, the sentences were not the main reason for the wrong translation. It seems more like the traditional problem of error-prone word alignment, affecting the translation of the sentence. Only 63 sentences participated in final translations; out of them, only 42 were translated correctly. Seeing these results, one can conclude that, unsurprisingly, the system is making bad choices when it tries to select the best fragments for incorporation in the final translations. Remember that our current example-based system is using a similarity score model to determine the similarity of sentence, and it still needs to be adapted to use a more standard model.

We expect that using this corpora, would result in even better performance. However, the parallel corpora we could find so far, that pair English with Swahili, contain a very limited quantity of sentences, which makes it irrelevant for the similar sentence extraction.

According to Kfir Bar et AL, proposed that it could have been easier to acquire accurate parallel corpora from UN published document Since Swahili and English are UN official languages, we could have built such corpora using the formal published documentation by the UN, provided in these languages

We based our idea on the work by Junsheng and sun, on Sentence similarity. Where he indicated similarities between words in different sentences have great influence on the similarity between two sentences. Words and their orders in the sentences are two important factors to calculate sentence similarity.

They also proposed three ways of measuring sentence similarity one is

- *Symbolic similarity.* If two sentences are more similar, then words in the two sentences are more similar, and vice versa. Here, words in two sentences are similar means that the words are similar in symbolic or in semantics.
- *Semantic similarity*. Two sentences with different symbolic and structure information could convey the same or similar meaning. Semantic similarity of sentences is based on the meanings of the words and the syntactic of sentence.
- *Structure similarity*. If two sentences are similar, structural relations between words are similar, and vice versa. Structural relations include relations between words and the distances between words. If the structures of two sentences are similar, they are more possible to convey similar meanings.

This study did not focus on semantic and syntactic analysis of sentences extracted. We chose to use symbolic similarity and supplement it with levenshtein edit distance to improve on its performance where distance between two strings source sentence(primary ) and target(translated) is defined as the minimum number of operations performed in order to obtain target(translated) from source sentence(primary ) (Levenshtein algorithm). The measure gives an indication for the closeness of the two strings. According to Stavros in his work indicated that we can solve the word difference problem by Computing the edit distance between two given words. The problem can be solved using a dynamic programming algorithm and also that the error-correction problem, given a language of valid (or correct) words, we can correct a given word to some word of the language that is the least different to the given word. This problem presupposes an agreed measure of word difference such as the edit distance.

Example based machine translation is a variant of corpus based MT, and is based on the following ideas.

- Humans do not translate a simple sentence by doing deep linguistic analysis rather,

- Humans translate by properly decomposing an input sentence into certain fragments. The translation of each fragment is then performed by the analogy translation principle, via proper examples.

An example based machine translation (EBMT) system retrieves similar examples (pairs of source phrases, sentences, or texts and their translations) from a database of examples (translation memory).which according to Friedel et al (2014), A translation memory system stores a data set of source-target pairs of translations. It attempts to respond to a query in the source language with a useful target text from the data set to assist a human translator. Such systems estimate the usefulness of a target text suggestion according to the similarity of its associated source text to the source text query. This study analyses two data sets in two language pairs each to find highly similar target texts, which would be useful mutual suggestions. We further investigate which of these useful suggestions cannot be selected through source text similarity, and we do a thorough analysis of these cases to categorize and quantify them. This analysis provides insight into areas where the recall of translation memory systems can be improved. Specifically, source texts with an omission, and semantically very similar source texts are some of the more frequent cases with useful target text suggestions that are not selected with the baseline approach of simple edit distance between the source texts the system helped in improving the performance of the translation system.

# Chapter 5: Conclusion

## 5.1 Contribution of this research

The system developed has demonstrated a promising potential for using sentence similarity in an example-based machine translation approach for English - Swahili, in particular. We found that sentence has provided better performance from being matched carefully by considering the topic of the sentence in which they are categorized.

Through this system we able to solve the problem of consistency in translation by using these tools based on translation memories. When using this system the translators is able to translate the text interactively with a computer tool that stores and retrieves all identical source sentences with their corresponding translations, guiding the translator towards consistent translations. We also made it possible to reuse old translations stored as translation memories of previous versions of handbooks and thereby minimizing the chances of producing variant translations of the same source sentence improving on the quality of the translation memories that are being put to use in a translation project.

Another interesting observation is the fact that using sentence similarity on a large corpus did not result in a significant improvement of the final results, as it did for a smaller parallel corpus. This suggests that sentence similarity can contribute to EBMT for language pairs lacking large parallel corpora, by enabling the system to better exploit the small number of examples in the given parallel corpus.

## 5.2 Challenges

A parallel corpus is an indispensable resource for work in machine translation and other multilingual NLP tasks. For some language pairs (e.g English, French, Swahili) data are plentiful. For most language pairs, however, parallel corpora are either nonexistent or not publicly available, and producing a domain-specific multilingual database was a tough, time consuming and complicated task, other limitations include copyright restrictions on data published.

This study did not focus on semantic and syntactic analysis of sentences extracted. Also using materials from the Web for language learning is risky, since there are sites where language is not used in a standard style.

Huangfu Wei (2013) concluded that the EMBT method needs a considerably large reference corpus with bilingual sentence examples, which need huge human and financial resources to build. If the coverage and size of the parallel corpus is limited, computers cannot find perfect matches for the translation. Moreover, the sentence similarity calculation is the bottleneck of this method because none of the available method can account for all the differences between languages and it is for these reasons that the EMBT method cannot be used to replace human translation and only as a plug-in method to improve efficiency and quality of transition. More work is still needed for better aligning the translation examples. Sometimes, even if the system succeeds in matching examples based on sentence similarity, the final translation was wrong due to alignment or sentence matching for the retrieved translation example. Of course, improving on the accuracy for the output translations is an essential step toward releasing the real potential of our system. This process is currently being investigated and planned for implementation in the near future.

Though the performance achieved by our system remains low, primarily because of the above-mentioned alignment and data accuracy issues, a detailed review of various translations suggests that the benefits of using matches based on sentence similarity will carry over to more complete translation systems. What is true for our automatically-generated sentences is even more likely to hold when a quality Swahili lexicon will become available for translation use. In the meanwhile, we will continue working on different methods for automatic extraction of sentence equivalents for English to Swahili.

From our system, the tool, which we used to find new sentences in a parallel corpus of comparable html documents, performs pretty well in terms of precision. We are now working on different techniques for building a new classifier for extracting sentences equivalents from a corpus of comparable html documents.

## 5.3 Future Research Avenues

These systems were chosen on the basis of their ability to process large quantities of text and were not selected on the quality of translations produced. This experiment could be extended by combining resources derived from alternative on-line EBMT systems. Dummy subjects could also be applied to derive sentence from other than third person singular and plural.

Comparing other ways of using sentence similarity matching also the issue of ambiguous sentences definitely is a direction for future investigation.

We even plan to consider more features. For instance – text to speech and since there are some recent works in this area, we will be considering incorporating such tools in future.

## Chapter 6: References


1. Rudi L. Cilibrasi and Paul M.B. Vit´anyi 1 Normalized Web Distance and Word Similarity
2. Rebecca Bruce M.S., M.S. 1995: A statistical method for word sense disambiguation. New Mexico State University Las Cruces New Mexico 1995.
3. Jose Joao Almeida, Alberto Manuel Sim~oes, and Jose Alves de Castro, 2005: Grabbing parallel corpora from the Web,Departamento de Inform_atica,Universidade do Minho.
4. Eneko Agirre and Oier Lopez de Lacalle, 2008:On Robustness and Domain Adaptation using SVD for Word Sense Disambiguation ,Informatika Fakultatea, University of the Basque Country 20018, Donostia, Basque Country.
5. Jordan Boyd-Graber, David Blei and Xiaojin Zhu, 2007: A Topic Model forWord Sense Disambiguation Computer Science, Princeton University and University of Wisconsin.
6. Gumwon Hong, Chi-Ho Li, Ming Zhou and Hae-Chang Rim, 2010:An Empirical Study on Web Mining of Parallel Data,Department of Computer Science & Engineering,Korea University Natural Language Computing Group,Microsoft Research Asia.
7. Delphine Bernhard, 2006:Multilingual Term Extraction from Domain-specific Corpora Using Morphological Structure,TIMC-IMAG Institut de l'Ing´enierie et de l'Information de Sant´eFacult´e de M´edecine F-38706 LA TRONCHE cedex.
8. Bo Li and Juan lee, Mining Chinese-English Parallel Corpora from the Web. School of Computer Science Wuhan University Wuhan, 430072, China.
9. Masao Utiyama, Daisuke Kawahara, Keiji Yasuda, and Eiichiro Sumita, 2009: Mining Parallel Texts from Mixed-Language Web Pages,National Institute of Information and Communications Technology (NICT), Keihanna Science City 619-0288 Kyoto, Japan.
10. Franz Josef Och and Hermann Ney, 2000: improved statistical Alignment Models. Computer science department RWTH Aachen – University of Technology D-52056.
11. Maarten van Gompel , Antal van den Bosch and Peter Berck, 2000: Extending Memory-Based Machine Translation to Phrases .ILK Research Group, Tilburg centre for Creative Computing Tilburg University, Tilburg, The Netherlands.
12. Uwe Reinke,2000: Towards a Closer Integration of Termbases, Translation Memories, and Parallel Corpora:-A Translation-Oriented View- Institute for Applied Linguistics, Translation and Interpreting Saarland University.

13. Hao-chun Xing (EMIS), Xin Zhang (EMIS), 2008. Using parallel corpora and Uplug to create aChinese- English dictionary. Department of Computer and Systems Sciences Stockholm University / Royal Institute of Technology, December 2008.
14. John Fry,1995: Assembling a parallel corpus from RSS news feeds, Artificial Intelligence Center SRI
15. Al o Gliozzo and Carlo Strapparava 2006: Exploiting Comparable Corpora and Bilingual Dictionaries for Cross-Language Text Categorization ITC –Irst via So;mmarive, I-38050, Trento, ITALY
16. Rodolfo Raya 2004: multi-platform translation/localisation and content publishing tools using XML and Java technology. Maxprograms' CTO (Chief Technical Officer),
17. Anil Kumar Singh and Samar Husain 2007:Exploring Translation Similarities for Building a Better Sentence Aligner Language Technologies Research Centre International Institute of Information Technology
18. Palakorn Achananuparp, Xiaohua Hu, and Shen Xiajiong 2008:The Evaluation of Sentence Similarity Measures College of Information Science and Technology Drexel University, Philadelphia, PA 19104 ,College of Computer and Information Engineering, Henan University, Henan
19. Yee Seng Chan and Hwee Tou Ng, 2009: MAXSIM: A Maximum Similarity Metric for Machine Translation Evaluation,Department of Computer Science National University of Singapore Law Link, Singapore 117590
20. Rejwanul Haque, Sudip Kumar Naskar, Andy Way CNGL, R. Costa-juss and Rafael E.2010 : Sentence Similarity-Based Source Context Modelling in PBSMT, School of Computing
21. AMINUL ISLAM and DIANA INKPEN 2008: Semantic Text Similarity Using Corpus-Based Word Similarity and String SimilarityUniversity of Ottawa
22. Andriy Nikolov, Victoria Uren, Enrico Motta and Anne de Roeck 2008 :Hand ling instance corefer encing in the KnoFuss architecture,Knowle dge Me dia Institute, The Open University, Milton K eynes , UK
23. Remco Dijkman ,Marlon Dumas, and Luciano Garcia-Banuelos 2001 :Graph Matching Algorithms for Business Pro cess Mo del Similarity Search,Eindhoven University of Technology, The Netherlands and University of Tartu.
24. Thanh Dao 2002: An improvement on capturing similarity between strings

25. R. Mihalcea, C. Corley, and C. Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. AAAI 2006, 6.
26. V. I. Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. Ph.D. thesis, Soviet Physics Doklady.
27. Masao Utiyama and Hitoshi Isahara 2003, Reliable Measures for Aligning Japanese-English News Articles and Sentences. Communications Research Laboratory, Proceedings of the 41st Annual Meeting of the Association 3-5 Hikari-dai, Seika-cho, Souraku-gun, Kyoto 619-0289 Japan
28. L i, y., m clean, d., bandar, z., o'shea,j.,and crockett, k. 2006. Sentence similarity based on semantic nets and corpus statistics. IEEE Trans. Knowl. Data Eng. 18, 8, 1138–1149.
29. Fernando Casanovas Martín, 2009: Approximate string matching algorithms in art media archives, AGH University of Science and Technology Faculty of Electrical Engineering, Automatics, Computer Science and Electronics
30. Jiannan Wang Guoliang Li Jianhua FengFast-Join 2011: An Efficient Method for Fuzzy Token Matching based String Similarity Join.Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, Beijing 10084, China
31. Kevyn Collins-Thompson and Jamie Callan 2007. Automatic and human scoring of word definition responses**.** Language Technologies Institute ,School of Computer Science Carnegie Mellon University.Pittsburgh, PA, U.S.A. 15213-8213.
32. Magnus Merkel 2003, checking translations for inconsistency – a tool for the editor . nlplab. Department of Computer and Information Science, Linköping University, Sweden
33. Karl-Johan Lönnroth 2005, Translation Tools and Workflow, Directorate-General for Translation, European Commission
34. Lagoudaki, E. 2006: Translation Memories Survey, London: Imperial College from:http://www3.imperial.ac.uk/portal/pls/portallive/docs/1/7294521,.
35. Timothy Baldwin and Hozumi Tanaka :The Effects of Word Order and Segmentation on Translation Retrieval Performance,Tokyo Institute of Technology 2-12-1 Ookayama, Meguro-ku, Tokyo 152-8552 Japan

36. Baker, M. 1993. Corpus Linguistics and Translation Studies – Implications and Applications. In Text and Technology, edited by M. Baker, G. Francis and T.-B. E. Philadelphia/Amsterdam: John Benjamins Publishing Company.
37. Magnus Merkel, 1996..Checking translations for inconsistency – a tool for the editor NLPLAB Department of Computer and Information Science Linköping University Sweden
38. Matthew R. Scott, Xiaohua Liu, Ming Zhou 2011. Microsoft Engkoo Team. Engkoo: Mining the Web for Language Learning. ACL
39. Kfir Bar and Nachum Dershowitz 2007: Using Semantic Equivalents for Arabic to English Example -Based Translation,School of Computer Science, Tel Aviv University, Ramat Aviv, Israel
40. Stavros Konstantinidis: language Computing the edit distance of a regular Department of Mathematics and Computing Science Saint Mary's University Halifax, Nova Scotia B3H 3C3, Canada s.konstantinidis@smu.ca
41. Friedel Wolff, Laurette Pretorius, Paul Buitelaar 2014: Missed opportunities in translation memory matching College of Graduate Studies University of South Africa 2INSIGHT Center for Data Analytics National University of Ireland, Galway wolfff@unisa.ac.za, pretol@unisa.ac.za, paul.buitelaar@deri.org
42. Huangfu Wei,2013: Investigating Core Technologies in Computer-aided Multi-lingual Translation Memory School of Foreign Languages, North China Electric Power University, Beijing 102206, China mailtohf@163.com
43. Nielsen, J. (1993). "Iterative User Interface Design". *IEEE Computer vol.26 no.11 pp 32-41*.

## Appendix

```
package com.drops.searchMaster;
import com.drops.entities.Finalmatch;
import com.drops.entities.Translatedsentences;
import java.util.ArrayList;
import java.util.Collections;
import java.util.LinkedList;
import java.util.List;
import java.util.Vector;
import javax.persistence.EntityManager;
import javax.persistence.NoResultException;
import javax.persistence.Persistence;
import javax.persistence.Query;
import javax.swing.table.AbstractTableModel;
import org.openide.util.Exceptions;
public class DropsModelMaker extends AbstractTableModel {

    public long srcLangID;
```

```
    public long transLangID;
    public String category;
    List<String> results = new ArrayList<String>(100);
    private EntityManager entityManager;
    private String title = "***_____________TRANSLATION_____________*****";

    public DropsModelMaker(String srcData, long srcLangId, long transLangId, String topic) {
        try {
            this.srcLangID = srcLangId;
            this.transLangID = transLangId;
            if (topic.equals("all")) {
                category = "";
            } else {
                category = topic;
            }
            entityManager                                                                    =
Persistence.createEntityManagerFactory("translationgeniusEntitiesPU").createEntityManager();
            //Check if sentence is already stored
            //No well translated data Hence use search Engine.
            DropsSearcher srch = new DropsSearcher();

            List<searchResults> search = srch.search(srcData,  category);

            if (!search.isEmpty()) {

                results.removeAll(results);
                int i = 0;
                for (searchResults rst : search) {

                  try{
                   if( rst.type.equals("final")){
                     String res = "SELECT f FROM Finalmatch f WHERE f.id = ?1";
                     List          resultList          =          entityManager.createQuery(res).setParameter(1,
rst.id).getResultList();
```

```
                Finalmatch r = (Finalmatch) resultList.get(0);
                rst.setTranslation(r.getTranslateddata());
              }else if(rst.type.equals("raw")){

                String res = "SELECT t FROM Translatedsentences t WHERE t.rawsentenceid = ?1";
                List resultList = entityManager.createQuery(res).setParameter(1,
rst.id).getResultList();
                                    Translatedsentences t =  (Translatedsentences) resultList.get(0) ;
                rst.setTranslation(t.getData());
}
              results.add(rst.translation);
            }catch(NoResultException ex){
              continue;
            }
            }
          }else {
            title = "The genius has no suggestions...";
          }

      } catch (Exception ex) {
        Exceptions.printStackTrace(ex);
      }
    }
    @Override
    public int getRowCount() {
      return results.size();
    }
    @Override
    public int getColumnCount() {
      return 1;
    }
    @Override
    public Object getValueAt(int row, int column) {
```

```
        if(row <0)
            return "";
        return results.get(row);
    }
    @Override
    public String getColumnName(int col) {
        return title;
    }
}
package com.drops.searchMaster;
public class searchResults {
        public int id;
    public float score;
    public String type;
    public String translation;
    public searchResults(int id, float score, String type) {
        this.id = id;
        this.score = score;
        this.type = type;
    }
    public void setTranslation(String translation) {
        this.translation = translation;
    }
```