# English-to-Sanskrit Statistical Machine Translation with Ubiquitous Application

Sandeep R. Warhade
Research Scholar
IT Deptt., Bharati Vidyapeeth Deemed University College of Engineering,
Pune-India.

Prakash R. Devale
Professor & Head,
IT Deptt.,Bharati Vidyapeeth Deemed University College of Engineering,
Pune-India.

Suhas H. Patil
Phd, Professor & Head,
Comp. Deptt., Bharati Vidyapeeth Deemed University College of Engineering,
Pune-India.

## ABSTRACT

In this paper by utilizing the capabilities of modern ubiquitous operating systems we introduce a comprehensive framework for a ubiquitous translation and language learning environment for English to Sanskrit Machine Translation. We present an application for learning Sanskrit characters, sentences and English Sanskrit translation. For the implementation, we have used the open-source Android platform on the Samsung Mini2440, a state-of-the-art development board. We present our current state of implementation, the architecture of our framework,and the findings we have gathered so far. In addition to this, here we describes the Phrase-Based Statistical Machine Translation Decoder for English to Sanskrit translation in ubiquitous environment. Our goal is to improve the translation quality by enhancing the translation table and by preprocessing the Sanskrit language text .

## General Terms

Artificial intelligence, machine learning, machine translation, statistical machine translation.

## Keywords

English-to-Sanskrit Translation , Ubiquitous Translation, Ubiquitous Computing, Statistical Machine Translation.

## 1. INTRODUCTION

Due to the growing usage of mobile devices, the concept of mobile and ubiquitous computing is becoming an increasingly important part of our everyday life because of their increasing computational power, quite large storage capacity, easy user interface and an improving network infrastructure. To support our daily life activities and provide various forms of entertainment, , there is a growing demand for mobile applications. For modern education system applications are required for translation and language learning software. There is often a limit to usefulness and operations of existing Web-based applications and a recently growing number of cell phone applications offering these services. Even though having the large capabilities of todays cell phones, they rarely utilize the sensors e.g. adjusting the noise level for learning material difficulty, for offering situation and location based learning suggestions, which affects the ability of the student to concentrate, or to consider the GPS position. Currently there is a lack of situational awareness and thereby important aspects of ubiquitousness.

The correlation of the important terms in a computer assisted learning context are demonstrates in [18] . The software or Web content accessible from a local workstation is the computer assisted learning. For the students it is easier to carry around this program and access it anywhere, e.g. utilizing a PDA, Tablets, Smart phones, in other way it is a mobile learning application. If the device and the applications are able to adjust and/or measure to the environment of the students then it is added pervasive component . A combination of mobile learning and pervasive is called ubiquitous learning and is defined by a high degree of mobility and embeddedness. The system which uses information about the current situation to refine or expand the translation, make suggestions for useful terms, sentences, names, etc is the ubiquitous translation system as a mobile translation system.

A device which offers portability, computational power, sensory capabilities, and ease of use, is the proper platform which transform this theory into a real application . The modern cell phone, PDA, Tablets completes all those prerequisites and is, beyond that, spread everywhere. As Mark Weiser mention that :

“The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.” [3]

Todays smart devices are on the verge of becoming such a profound technology. The usability of smart devices with Android operating system is improving through growing wireless telecommunication capabilities, increasing network coverage, and enhanced battery technology. Todays our small hand-held device application range is expanding tremendously. Embedded devices specially cell phones with integrated browsers, vast animation/graphics functions and GPS localization are no exception any more and their applications are becoming more powerful and more user friendly. By maintaining the size and capabilities of a regular cell phone, recent merging of Internet tablets and cell phones undertaken by various mobile phone manufacturers, is a good example of a big step towards a device which can handle communication and computation tasks of a desktop computer. Like on a desktop computer this enables us to access information from wherever we are and whenever we want and process this information. However, compared with a desktop computer we have less hardware resources but additional sensors attached to it, such as GPS position calculation, noise levels measured by the microphone, and acceleration data measured by the phone’s movement sensors. Furthermore, there is a smaller screen to display all this information to the user. Hence, we have to put much effort into the design of the user interface and the architecture regarding computation times. The primary reason why we have decided to develop on the development board, is the fact that its native operating system, Android, is a Linux-based distribution and amongst other advantages .

In this paper we present our research on how a cell phone operating system can be utilized and how its capabilities can be exploited to provide the best possible language learning and translation services, leading to a ubiquitous language learning and translation environment. We report on our implementation experience, which we have gathered so far, and describe some of our previous research . which we worked to create the ubiquitous translation and language learning environment . The system is modular and flexible, and an adjustment or extension to other languages would be a matter of changing mere implementation details and adding the language-specific resources, such as lexical, parser, corpora, etc.

Automatic translation from one natural language into another using computers is the Machine Translation. Statistical machine Translation is an approach to MT that is characterized by the use of machine learning methods. This means that we apply a learning algorithm to large body of previously translated text, known variously as a parallel corpus, parallel text, bi-text or multi-text. With an SMT toolkit and enough enough parallel text, we can build an MT system for a new language pair within a very short period of time. The accuracy of these systems depends crucially on the quantity, quality and domain of the data. Making a sequence of word translation and reordering decisions perform translation. Word translation is often ambiguous, means it is common for the different possible translations of a word to have very different meanings. Often the correct choice will depend on context. Therefore, our system will need some mechanism to correctly reorder the words. Reordering is typically dependent on the systematic structure of the target language. As with word translation,reordering decisions often entail resolving some kind of ambiguity. English is a well known language so we illustrate Sanskrit grammar and its silent features. The English sentence always has an order of Subject-Verb-Object, while Sanskrit sentence has a free word order. A free order language is a natural language which does not lead to any absurdity or ambiguity thereby maintaining a grammatical and semantic meaning for every sentence obtained by the change in the ordering of the words in the original sentence. For example , the order of English Sentence (ES) and itself equivalent translation in Sanskrit Sentence (SS) is given as below.

| | | | |
|---|---|---|---|
| ES: | Ram<br>(Subject) | reads<br>(Verb) | book.<br>(Object) |
| SS: | Raamah<br>(Subject) | pustkam<br>(Object) | pathati.<br>(Verb) |
| | Pustkam<br>(Object) | raamah<br>(Subject) | pathati.<br>(Verb) |
| | Pathati<br>(Verb) | pustkam<br>(Object) | raamaha.<br>(Subject) |

Thus Sanskrit sentence can be written using SVO, SOV and VOS order.

The rest of this paper is organized as follows. In Section “RELATED WORK” we discuss research relevant to our work. We then describe the system architecture of our translation and language learning environment in Section “SYSTEM ARCHITECTURE”. We elaborate on our earlier research efforts, which we have integrated into this work, in Section “SMT SYSTEM FRAMEWORK”, and we conclude the paper with a summary and an outlook in Section “CONCLUSION”.

## 2. RELATED WORK

There is a different types of Web-based translation services and language learning applications available, e.g. [14, 7, 21]. There are different approaches of translation service techniques from machine translation. An overview of those can be obtained from [10]. Based on [14, 8, 12] we have found that our method, being example and corpus based, is well suited for a language learning application, since it contains intermediate information in the translation processes, which can be valuable for a language student. The ubiquitous property of mobile devices has been successfully used to offer contextual learning environments in the past [9,13,11,17]. The application of ubiquitous learning demands several characteristics to be fulfilled [15]:

• Permanency: The entire learning process is recorded continuously. In addition, Learners never lose their work unless it is purposefully deleted.

• Accessibility: Based on learners requests they have access to their documents, data, or videos from anywhere. Therefore, the learning involved is self-directed.

• Immediacy: Learners can get any information immediately wherever they are. Thus, learners can solve problems quickly. It solves query recording and look for the answer later.

• Interactivity: The experts are more reachable and the knowledge becomes more available. Learners can interact with experts, teachers, or peers in the form of synchronous or asynchronous communication.

• Situation of instructional activities: The problems encountered as well as the knowledge required are all presented in their natural and authentic forms. The learning could be embedded in our daily life. This helps learners to notice the features of problem situations that make particular actions relevant.

An example of the application of those guidelines can be seen in [1]. The idea behind it is to give the students the chance to efficiently use their time and the ability to access class room information at will. In [2] a ubiquitous learning environment was developed by using IEEE 802.11 WLAN and Bluetooth for network communication. This showed that a learning experience, supported by a contextually matching surrounding, is more valuable in terms of understanding and memorizing since it is based on an inductive process. However, the limits of the network technologies do not allow a deployment of that system into a large network structure. As pointed out by [5], language learning is a life-long activity, and support by ubiquitous learning environments can accompany the learner at all stages. Language learning takes place virtually anywhere and is optimized if supplemented on demand. The need for immediate help makes translation resources on mobile devices very valuable, which is the reason we have focused on combining those two, so that they complement each other in the best way possible.

## 3. SYSTEM ARCHITECTURE

The framework is designed as a client-server setup. The client is the development board, in our case the FriendlyARM Mini 2440 SBC (Single-Board Computer) with 400 MHz Samsung S3C2440 ARM9 processor. The LAMP (Linux-Apache-

MySQL-PHP) server is situated at the Bharti Vidyapeeth College of Engineering, Pune. The hardware specifications of both are listed in Table 1.

**Table 1. Hardware Configuration**

| | **Server** | **Client**-Mini 2440 SBC |
|---|---|---|
| CPU | 4x Intel(R) Xeon(R) CPU E5405 @ 2.00GHz | 400 MHz Samsung S3C2440A ARM920T |
| RAM | 4057MB | 64 MB |
| OS | 4057MB | 64 MB |
| Sec. Memory | 1.7 TB | 1GB |

The operating system on the development board is Android, the open-source , running on a Linux kernel, designed for smart phones and Internet tablets. Android is a software stack for mobile devices that includes an operating system, middleware and key applications. It is developed by the Open Handset Alliance , led by Google, and other companies. The Open Handset Alliance is a group of mobile operators, hardware manufacturers, semiconductor companies, software companies and commercialization companies. For development we have used Eclipse, a editor, emulator and cross-compilation toolkit , in combination with Java. In contrast to distributions for desktop computers, Android has touch screen support, sliding keyboard support, an interface to the Camera, GPS, compass, and accelerometer, while discarding some typical desktop distribution functionalities. Even though the hardware on the client side is quite powerful for a cell phone, it is not enough to perform all the calculations needed for our framework in a reasonable amount of time. Hence, we need to outsource as many calculations as possible to the server, where they can be processed much quicker. We have considered the fact that a cellular phone is not always guaranteed to have a decent network connection. Spots without a carrier signal, such as tunnels, elevators, etc. have to be taken into consideration. Therefore, the communication between the server and the client is done over asynchronous calls, to guarantee a service even in the case of an interruption of the network connection. Additionally, a database on the mobile device is kept to store user input, such as vocabulary lists or program preferences, and enable system use while the network and/or the server is not available. Outdated entries from this database are purged periodically, due to limited storage capabilities on the cell phone.

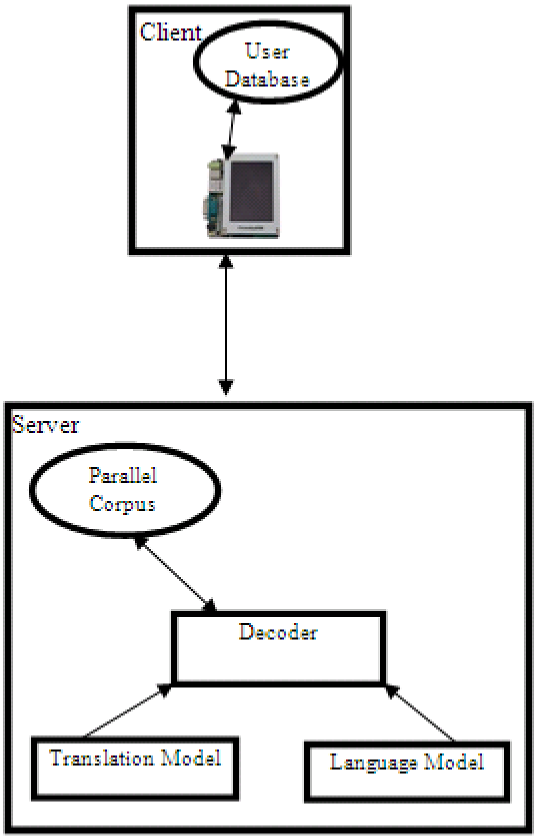

**Fig 1: System Architecture**

The overview of the server-client architecture is shown in Fig. 1. The translation module presents the contents graphically and provides input fields. The language learning module offers a graphical learning program with various functionality. Data from the knowledge base on the server are used to construct a customized learning support, in terms of word/sentence suggestions and difficulty level. Both inter faces store essential data on the device. The language learning module accesses the personal database on the phone, which holds the necessary information to operate the framework without a network connection, though with limited capabilities. This personal database is synchronized with the user-specific database on the server, whenever possible. The user-specific database, stores the user's history, such as previous query sentences. Words, compounds, and sentences stored in this database are assumed to be of interest to the student and are preferred in lecture suggestions.

## 4. SMT SYSTEM FRAMEWORK

Phrase-based statistical machine translation approaches continue to dominate the field of machine translation. The translation service makes use of state-of-the-art phrase based SMT systems within the framework of feature-based exponential models containing the following features:

a. Phrase translation probability

b. Inverse phrase translation probability

c. Lexical weighting probability

d. Inverse lexical weighting probability

e. Phrase penalty

f. Language model probability

g. Simple distance-based distortion model

h. Word penalty

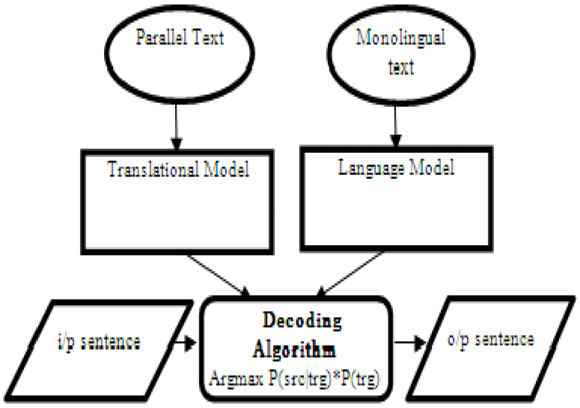

**Fig 2: Basic SMT framework**

The basic framework within which all the MT systems were constructed is shown in Figure 2. Translation examples from the respective bilingual text corpus are aligned in order to extract phrasal equivalences and to calculate the bilingual feature probabilities. Monolingual features like the language model probability are trained on mono lingual text corpora of the target language whereby standard word alignment and language modeling tools were used.

## Parallel Corpora

SMT treats translation as a machine learning problem. This means that we apply a learning algorithm to a large body of previously translated text, known variously as a parallel corpus, parallel text, bi-text, or multi-text. The learner is then able translate previously unseen sentences. With an SMT toolkit and enough enough parallel text, we can build an MT system for a new language pair within a very short period of time. be built for a wide variety of language pairs within similar time frames. The accuracy of these systems depends crucially on the quantity, quality, and domain of the data, but there are many tasks for which even poor translation is useful [20].

## Language Model

Statistical language modeling is the science (and often art) of building models that estimate the prior probabilities of word strings. Language modeling has many applications in natural language technology and other areas where sequences of discrete objects play a role, with prominent roles in speech recognition and natural language tagging (including specialized tasks such as part-of-speech tagging, word and sentence segmentation, and shallow parsing). As pointed out in [6], the main techniques for effective language modeling have been known for at least a decade, although one suspects that important advances are possible, and indeed needed, to bring about significant breakthroughs in the application areas cited above—such breakthroughs just have been very hard to come by [23,16].

## Translational model

Translational models allow us to enumerate possible structural relationships between pairs of strings. However, even within the constraints of a strict model, the ambiguity of natural language results in a very large number of possible target sentences for any input source sentence. Our translation system needs a mechanism to choose between them. This mechanism comes from modeling: parameterization. We design a function that allows us to assign a real-valued score to any pair of source and target sentences. The general forms of these models are similar to those in other machine learning problems. There is a vast number of approaches;, for more detail, the reader is referred to a general text on machine learning, such as [24].

## Decoding Model

Now that we have a model and estimates for all of our parameters, we can translate new input sentences. This is called decoding. We call this the decision rule. The phrase based decoder we developed for purpose of comparing different phrase-based translation models employs a beam search algorithm, similar to the one by [22].

The Sanskrit output sentence is generated left to right in form of partial translations (or hypotheses). We start with an initial empty hypothesis. A new hypothesis is expanded from an existing hypothesis by the translation of a phrase as follows: A sequence of untranslated foreign words and a possible English phrase translation for them is selected. The English phrase is attached to the existing English output sequence. The foreign words are marked as translated and the probability cost of the hypothesis is updated.

## Data flow of the system

As a translation basis we take the output of the statistical machine translation system Moses [4]. The overview of the dataflow in the system is shown in Fig. 3. An input sentence is sent to the part-of-speech (PoS) tagger for English and Sankirit as translation model. After each sentence token is assigned a PoS-tag, the sentence and its tags are compared with sentences from a preformatted corpus. For this purpose, we have modified and enriched a bilingual data collection consisting of 1500 sentences, taken from Sanskrit learning books. We have removed as much noise as possible from the data, assigned PoS-tags to each sentence token and stored the in formation in an SQL database. We have created different formats of the bilingual data, one with a complete set of PoS information and others with reduced and optimized tag sets to provide quick access and efficient processing. Additional representations and tag sets can be added easily to satisfy different needs in future work. We have applied relational sequence alignment [25] to obtain clusters of structurally similar sentences, so that the comparison of th query sentence with the clusters yields several similar structures. At the same time, the query sentence is processed with Moses to obtain a preliminary translation. This translation is then used to fill the template of the structures, which had been found to be similar in terms of PoS-tags. This way, a certain number of translation candidates is produced. The parameters of the similarity measure can be adjusted to fine-tune the result, depending on the text type and text domain. Allowing low threshold values for similarity, a higher number of candidates can be produced, whereas a higher threshold value reduces the number of candidates.

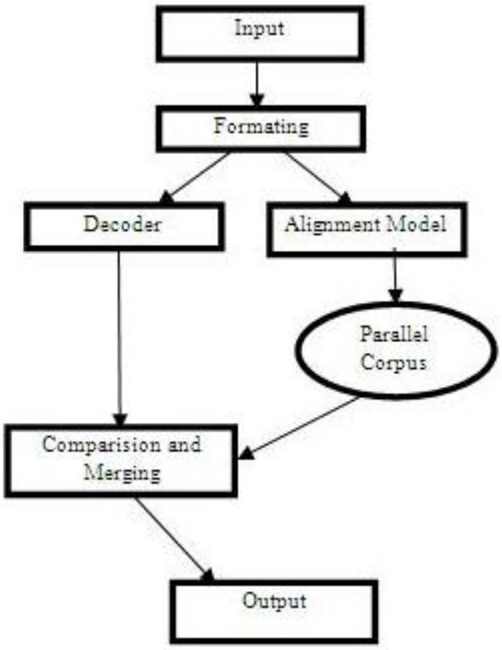

**Fig 3: Data Flow of System**

## 5. CONCLUSION

In this paper we have described the design and a design of a ubiquitous translation and language learning framework, in particular for English to Sanskrit, on the development board Mini 2440 SBC, a growing cellular phone operating system with internet capabilities. We have presented our implementation of a translation and language learning environment built as a client-server system, which consists of a translation and a language learning task. For the language learning task we have built a learning application . For the translation task we used statistical machine decoder, a translation framework. By integrating SMT decoder into this research work, we have shown how a client/server configuration can be realized to offer the entire translation service on the mobile device.

In future work, we want to focus on audio input, GPS localization, and user query statistics to create a detailed user profile. This will facilitate a specific fine-tuning of the learning environment, considering guidelines for computer enhanced learning such as the cognitive load theory, and the GUI design. As mentioned before, the already implemented learning environment is available on the Web to receive feedback from the Android community. In addition, we will make our framework available to students at our Language Department, once it is in a workable state. We will evaluate the framework with a sufficient number of users and a control group along the dimensions engagement, effectiveness, and viability [19], as well as analyze the usability on other prevalent mobile platforms, such as iPhone, Maemo5's successor MeeGo, etc.

## 6. ACKNOWLEDGMENTS

I like to extend my thanks to Prof. Emmanual M. HoD IT PICT who guided me throughout the preparation of the paper and gave me relevant concept to materialize the paper. I would also like give my humble regards to Prof. Girish Potdar Professor PICT for helping me to shape out the important findings and necessary reseach for the paper. I wish both the all the success for future.

## 7. REFERENCES

[1] B. Bomsdorf. Adaptation of learning spaces: Supporting ubiquitous learning in higher distance education. In N. Davies, T. Kirste, and H Schumann, editors, Mobile Computing and Ambient Intelligence: The Challenge of Multimedia, number 05181 in Dagstuhl Seminar, Dagstuhl, Germany, 2005.

[2] V. Jones and J. H. Jo. Ubiquitous learning environment: An adaptive teaching system using ubiquitous technology. In Proceedings of the 21st ASCILITE Conference, 2004.

[3] M. Weiser. The computer for the 21st century. SIGMOBILE Mob. Comput. Commun. Rev., 3(3):3–11, 1999.

[4] H. Hoang et al. Moses: Open source toolkit for statistical machine translation. Pages 177–180, 2007

[5] H. Ogata. Computer supported ubiquitous learning: Augmenting learning experiences in the real world. In IEEE International Conference on WireleLos Alamitos, CA, USA, 2008. IEEE Computer Society. ss, Mobile, and Ubiquitous Technology in Education, pages 3–10, Los Alamitos, CA, USA, 2008. IEEE Computer Society.

[6] P. Clarkson and R. Rosenfeld, ―Statistical language modeling using the CMU-Cambridge toolkit‖, in G. Kokkinakis, N. Fakotakis, and E. Dermatas, editors, Proc. EUROSPEECH, vol. 1, pp. 2707–2710, Rhodes, Greece, Sep. 1997.

[7] W. Winiwarter. WILLIE – a Web Interface for a Language Learning and Instruction Environment. In Proceedings of the 6th InternationalConference on Web-based Learning, Edinburgh, United Kingdom,2008. Springer-Verlag.

[8] W. Winiwarter. JETCAT – Japanese-English Translation using Corpus-based Acquisition of Transfer rules. JCP, 2(9):27–36, 2007

[9] C. Yin, H.Ogata, and Y. Yano. JAPELAS: Supporting Japanese polite expressions learning using PDA(s) towards ubiquitous learning. International Journal of Information and Systems in Education, 3(1):33–39, 2005.

[10] Y. Wilks. Machine Translation: Its Scope and Limits. Springer-Verlag, 2008.

[11] L. H. Gan et al. Language learning outside the classroom using hand helds with knowledge management. In

Proceedings of the 2007 Conference on Supporting Learning Flow through Integrative Technologies, pages 361–368, Amsterdam, The Netherlands, 2007. IOS Press.

[12] W. Winiwarter. WETCAT – Web-Enabled Translation using Corpus-based Acquisition of Transfer rules. In Proceedings of the Third IEEE International Conference on Innovations in Information Technology, Dubai, United Arab Emirates, 2006

[13] H. Ogata et al. Computer supported ubiquitous learning environment for Japanese mimicry and onomatopoeia with sensors. In Proceedings of the 2007 Conference on Supporting Learning Flow through Integrative Technologies, pages 463–470, Amsterdam, The Netherlands, 2007. IOS Press

[14] C. Goutte et al., editors. Learning Machine Translation. MIT Press,Cambridge, Massachusetts, 2009.

[15] Y. Chen et al. A mobile scaffolding-aid-base bird-watching learning system. In Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Eduction, pages 15–22. IEEE Computer Society Press, 2002.

[16] R. Rosenfeld, ―Two decades of statistical language modeling: Where do we go from here?‖, Proceedings of the IEEE, vol. 88, 2000.

[17] H. Ogata. Computer supported ubiquitous learning: Augmenting learning experiences in the real world. In IEEE International Conference on Wireless, Mobile, and Ubiquitous Technology in Education, pages 3–10, Los Alamitos, CA, USA, 2008. IEEEComputer Society.

[18] K. Lyytinen and Y. Yoo. Issues and challeDagstuhl Seminar, Dagstuhl, Germany, 2005. nges in ubiquitous computing. Commun. ACM, 45(12):62–65, 2002.

[19] D. A. Norman and J. C. Spohrer. Learner-centered education. Commun. ACM, 39(4):24–27, 1996.

[20] CHURCH, K. AND HOVY, E. 1993. Good applications for crummy machine translation. Mach. Transl. 8, 239–258.

[21] N. Nagata. Banzai: Computer assisted sentence production practice with intelligent feedback. In Proceedings of the Third International Conference on Computer Assisted Systems for Teaching and Learning/Japanese (CASTEL/J), 2002.

[22] Jelinek, F. (1998). Statistical Methods for Speech Recognition. The MIT Press.

[23] F. Jelinek, ―Up from trigrams! The struggle for improved language models‖, in Proc. EUROSPEECH, pp. 1037–1040, Genova, Italy,Sep. 1991.

[24] MITCHELL, T. M. 1997. Machine Learning. McGraw-Hill.

[25] A. Karwath and K. Kersting. Relational sequence alignments and logos. pages 290–304, 2007.