A COLLABORATIVE PLATFORM TO COLLECT DATA FOR DEVELOPING MACHINE TRANSLATION SYSTEM

BY

MD. ARID HASAN
ID: 151–15–5332

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering.

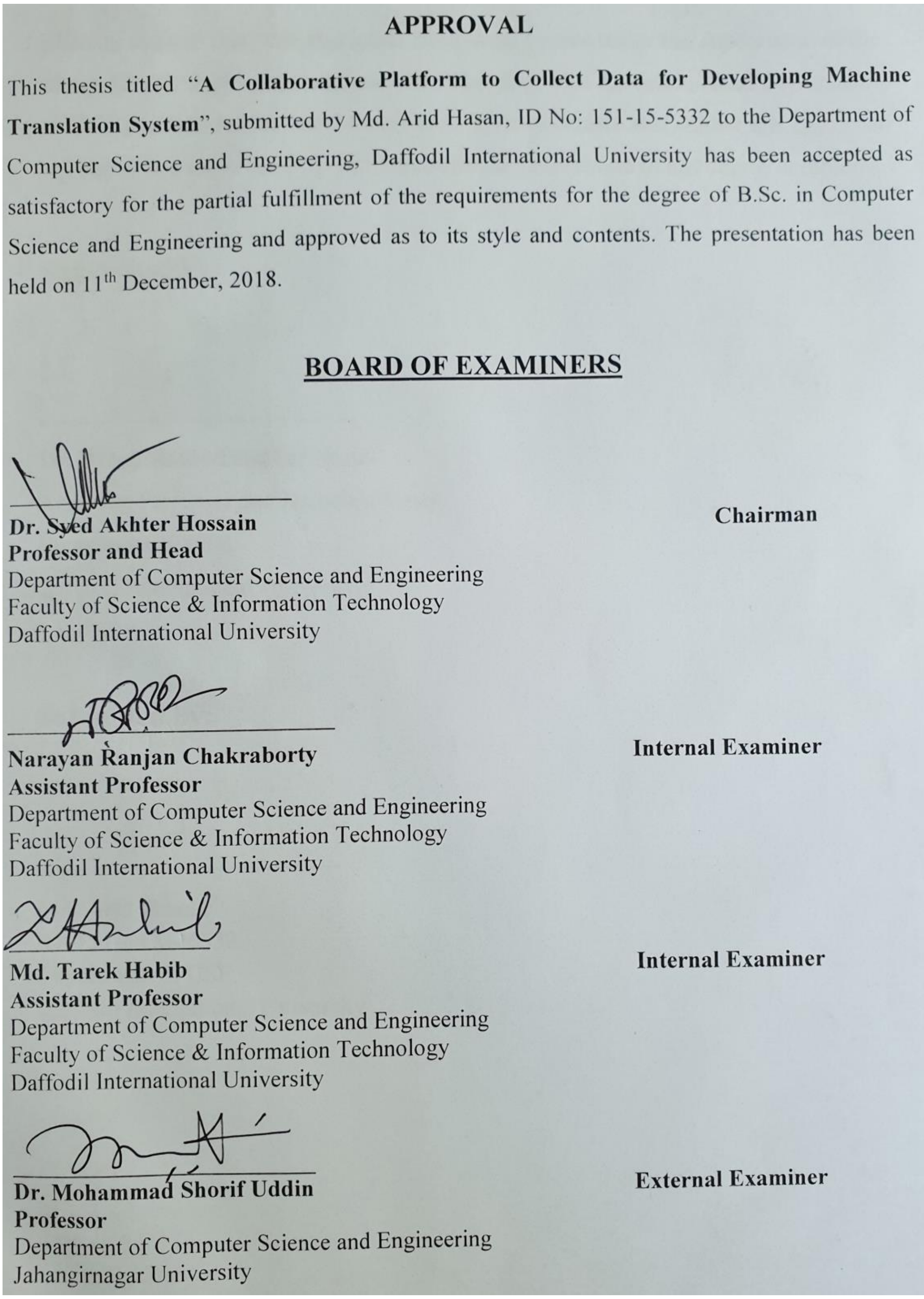
Supervised By

Sheak Rashed Haider Noori, PhD
Associate Professor and Associate Head
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2018



APPROVAL

This thesis titled "**A Collaborative Platform to Collect Data for Developing Machine Translation System**", submitted by Md. Arid Hasan, ID No: 151-15-5332 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 11th December, 2018.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Narayan Ranjan Chakraborty **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Tarek Habib **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Mohammad Shorif Uddin **External Examiner**
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

I hereby declare that, this thesis has been done by me under the supervision of **Dr. Sheak Rashed Haider Noori, Associate Professor and Associate Head Department of CSE,** Daffodil International University. We also declare that neither this thesis nor any part of this thesis has been submitted elsewhere for award of any degree or diploma.

Supervised by:

____

Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head
Department of CSE
Daffodil International University

Submitted by:

____

Md. Arid Hasan
ID: 151 – 15 – 5332
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

I have given my efforts to this thesis. However, it would not have been possible without the kind support and help of many individuals. I would like to express my deepest appreciation to all those who provided me the possibility to complete this report.

At first, I express my heartiest thanks and gratefulness to almighty Allah for His divine blessings which allowed me to complete this thesis successfully.

A special gratitude I give to my supervisor, Dr. Sheak Rashed Haider Noori, Associate Professor and Associate Head of CSE department, whose contribution in stimulating suggestions and encouragement, helped me to coordinate my thesis especially in writing this report. His endless patience, scholarly guidance, constant and energetic supervision, constructive criticism, valuable advice have made it possible to complete this thesis.

Furthermore, I would also like to acknowledge with much appreciation the crucial role of my department head, Professor Dr. Syed Akhter Hossain, who provided me with his precious time and kind help to finish this thesis. I also give my deepest thanks to all the faculty members and staff of CSE department of Daffodil International University.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

ABSTRACT

The emergence of neural machine translation techniques has opened up a new era for developing translation systems. However, it requires a very large amount of parallel corpus, which is scarce for many under-resourced languages, e.g., Bangla. In order to develop a corpus, currently, there is a lack of publicly available collaborative system. In this paper, we report an online collaborative system for the development of the parallel corpus. The system is developed for supporting any language, however, we only evaluated for developing Bangla-English parallel corpus. In a task completion evaluation experiment, the system outperforms the widely used offline system i.e., OmegaT.

Keywords: Machine Translation, Collaborative Platform.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

The significant progress of Machine Translation (MT) system has started back in the late 1980's and early 1990's due to the availability of computational power and parallel corpora. Notable research work include IBM statistical machine translation models [5], phrase-based MT models by Koehn et al [14], and hierarchical phrase-based statistical MT [6] among others. The development of open-source Statistical Machine Translation (SMT) toolkit Moses [15] facilitated the SMT research one step further. Due to the advancement of computational power and the notable work of deep learning, in the last few years, machine translation systems are moving forward with a new approach called Neural Machine Translation (NMT) [7].

From SMT approach to NMT approach, a common idea is to learn the parameters from a large amount of parallel corpus. It is clear that these techniques are highly dependent on the human translated aligned parallel corpus. One of the publicly available and a large corpus is Europerl [13], which has been extracted from the proceedings of the European Parliament consists of 21 European languages. Developing such a resource is difficult and challenging for under-resourced languages such as Bangla.

In order to develop the parallel corpus research community has been trying to develop tools such as OmegaT[1] and Zanata[2] to facilitate translators in their translation task. This translation task is highly memory intensive and time-consuming. Tools like OmegaT are not many and also poses some challenges. For example, OmegaT is an offline system, which lacks a collaborative translation mechanism, whereas zanata is an online system, however, it does not support dictionary like Bangla. In order to overcome these limitations, in this study, we develop a tool (i.e., named as *AmaderCAT*) for the translators, that is collaborative and highly configurable for the translation task.

[1] http://omegat.org

[2] http://zanata.org

It has the mechanism for crowd translation. In this paper, we report the implementation details of this tool and present some preliminary results.

1.2 Motivation

Bengali is usually counted as the seventh most spoken native language in the world by population. But we have low resources in the field of Natural Language Processing. Like, we have lack of enough publicly available parallel corpuses. Neural Machine Translation (NMT) requires millions of parallel sentences to get desired translation accuracy.

1.3 Rationale of the Study

In research area, incredible improvements of Machine Translation in few natural languages like English, Chinese, French, and Spanish etc. For low resources languages like Bangla, we don't have enough research work for Machine Translation. Lack of parallel corpuses our researcher are not willing to work for Bangla language. Bangla language has one of the complex grammar rules. This is one of biggest the reason that people are not willing to work. Those people who works for Bangla language they face lots of problem. One of them is, they can't work collaboratively in the online for collecting parallel sentences, Part-of-Speech tagging etc.

We are living in the era of Neural Network, where millions of data use for training to get desired output. The accuracy of desired output depends on how much data were used while training.

1.4 Outcome

Our main contributions include,

1) The development of the '*AmaderCAT'* tool, which will be open source and made publicly available[3],
2) The developed parallel corpus[4], which will be released for research purpose
3) Our preliminary experimental results using the corpus.

[3] https://github.com/AridHasan/Data-Collection-System-for-Machine-Translation

[4] https://github.com/AridHasan/Data-Collection-System-for-Machine-Translation/tree/master/data

1.5 Report Layout

In this chapter we have discussed about the introduction which shows Machine Translation and Collaborative Platform, motivation, rationale of the study and the outcome of the thesis. Later followed by the report layout.

In chapter 2, we will discuss about the state of the art of our research.

In chapter 3, we will discuss about the methodology of our entire thesis work.

In chapter 4, we will discuss about the requirement specification and collaborative platform development and its architecture.

In chapter 5, we will discuss about the data collection procedure.

In chapter 6, we will discuss about the system testing and system evaluation.

In chapter 7, we will discuss about the overall perception about our thesis and future work.

CHAPTER 2

STATE OF THE ART

In this Section, we first discuss the related work that has been done for developing parallel corpus. Then we discuss related work for Bangla Machine Translation.

In Table 2.1, we provide a selected list of tools that are commonly used for parallel corpus development. We listed them based on the license, online/offline and collaborative functionalities. Listing these tools helped use in understanding why it is important to develop a new tool, which can provide additional benefits to the research community.

Table 2.1: Most widely used parallel corpus development tools. Comm. - commercial.

Name	**License**	**Online/ Offline**	**Collaborative**
SDL Trados Studio [10, 20, 21, 24]	Comm.	Online/ Offline	Yes
OmegaT	GPLv3+	Offline	No
Zanata	GNU	Online	Yes
Sketch Engine	Comm./Free	Online	No

Typically, most of the translation systems use Translation Memory (TM) [5] [22], which facilitates the translation tasks by automatically suggesting the matched translation. The matching is typically done by fuzzy matching[6]. Translation memory systems actively support the translation process by automatically suggesting existing translations and terminology [12, 22]. TM helps the translation process faster by showing the meaning of a matching sequence in different sentences and the meaning

[5] TM is a database consists of source and target language pairs. It is typically stored in the database while translating the text corpus by the translators.

[6] Fuzzy matching is an approximate matching approach that tries to find a segment of matched translation by matching them with previously translated sentences. The segment can be a phrase or the whole sentence.

may be different from each other. It also increases the number of the translated sentence in Computer Assisted Translation System by assisting and accelerating the translation process [17].

The tools mentioned in Table 2.1, consists of different license such as commercial, and free and open-source. SDL Trados Studio is a commercial software while Sketch Engine provides both free and commercial support. Since OmegaT and Zanata are open-source, therefore, we have been interested to use and compare our system with these two tools. OmegaT has been one of the widely used tools [11] which also utilizes a Translation Memory (TM) while translating the sentences in order to facilitate the translators. At the same time, it uses a terminology, which also helps user while translating. The limitation of OmegaT is that it is an offline system, therefore, it is difficult to manage translations across translators. It also poses challenges while distributing the translation task for a large number of crowd translators. Zanata is another open-source and collaborative tool, however, the limitation is that it lacks dictionary support for the language other than English. An open source platform named OmegaT, used for Phrase based English – Tamil Translation System by Concept Labeling using Translation Memory [11]. OmegaT builds a Translation Memory while translating the sentences or receive Translation Memory from users before translating sentences.

In comparing with them, our proposed system uses verified translated sentences as Translation Memory. In addition, we use suffix striping using predefined rules. The TM checks all the possible match with fuzzy matching at runtime and shows all the matching source words with meaning to the translators. The suffix stripping checks the longest possible suffix and remove the unnecessary suffix from the root words, if exists and attach some vowels at the end of the words for getting actual root words and finds the meaning for the root word in the glossary and shows to the translators. We evaluated our system using Bangla-English parallel corpus development. However, this will work for any language pair.

The work related to the Bangla Machine translation system are relatively few compared to the other language. In [4], authors used Context-Free Grammars (CFGs) to develop English to Bangla Machine Translation system, which uses sentence construction rules for translating English to Bangla. The authors in [3] used transfer machine translation approach Bangla English machine translation system [3], which can translate simple assertive sentences. Ahmed et al. defined mapping rules for the development of MT system, which utilizes Bangla grammatical structure from an input English sentence and can translate affirmative sentences [1].

For Indo-Aryan languages, such as Bengali, Hindi with highly enrich grammar are difficult to find its morphological differences. A Rule Based Bengali Stemmer [18] used for decelerate the morphological diversity for a single word. Stemmer help to make better TM for the Bengali language. We build a Translation Memory (TM) which will help the translators showing a single word in different sentences with single or different meaning and a Glossary option to shows the meaning of a word/phrase to the translator which will reduces the time of translators to translate and more number of sentences can be translated in few moments. After translating the source sentences, a parallel corpus developed by the system which is ready for training.

CHAPTER 3
RESEARCH METHODOLOGY

The goal of this thesis is to understand the parallel corpus building with collaborative work and the uses of this corpus in Machine Translation using Neural Network. To successfully conduct the thesis below steps were taken.

- A study on existing system of parallel corpus building with collaborative work.
- Practical use cases in lingual diversity and various inflected form.
- Existing parallel corpus building systems which are provided currently is studied.
- Various papers on Machine Translation, Translation Memory (TM), Stemmer, Neural Network Algorithm and Cognitive Walkthrough were studied.
- To get the complete idea of the researchers' and translators need, the need of a survey is realized and justified.
- Various papers and books on survey methods were studied.
- The key factors which are needed to understand were identified, which are: machine translation, Translation Memory, glossary, Collaborative Platform and computer-assisted translation.
- The questions relevant to those facts were identified.
- Desired answers were divided into quantitative and qualitative data.
- Based on the key facts (machine translation, Translation Memory, glossary, Collaborative Platform and computer-assisted translation) relevant questionnaires were developed.
- Side by side, the survey setup and survey conduction plan were developed
- Finally, a complete set of questionnaires and survey conduction plan is proposed which ultimately will be used to understand the system usability of the researchers and translators

CHAPTER 4

REQUIREMENT SPECIFICATION AND SYSTEM ARCHITECHTURE

4.1 Use Case Modeling

For developing a system few techniques are used to collect the requirements. Use case modeling is one of those techniques along with UML class diagram, Communication diagram, activity diagram, etc. Use case is a sequence of actions to be performed to reach the goal of small part of a project. It’s also known as Unified Modeling Language as actor. It’s easy to represent the whole concept of the project using diagram. Use case diagram define the interactions between the system and the users. Normally three type of users can communicate with our system those are Admin, Expert Translators and Beginner Translators.

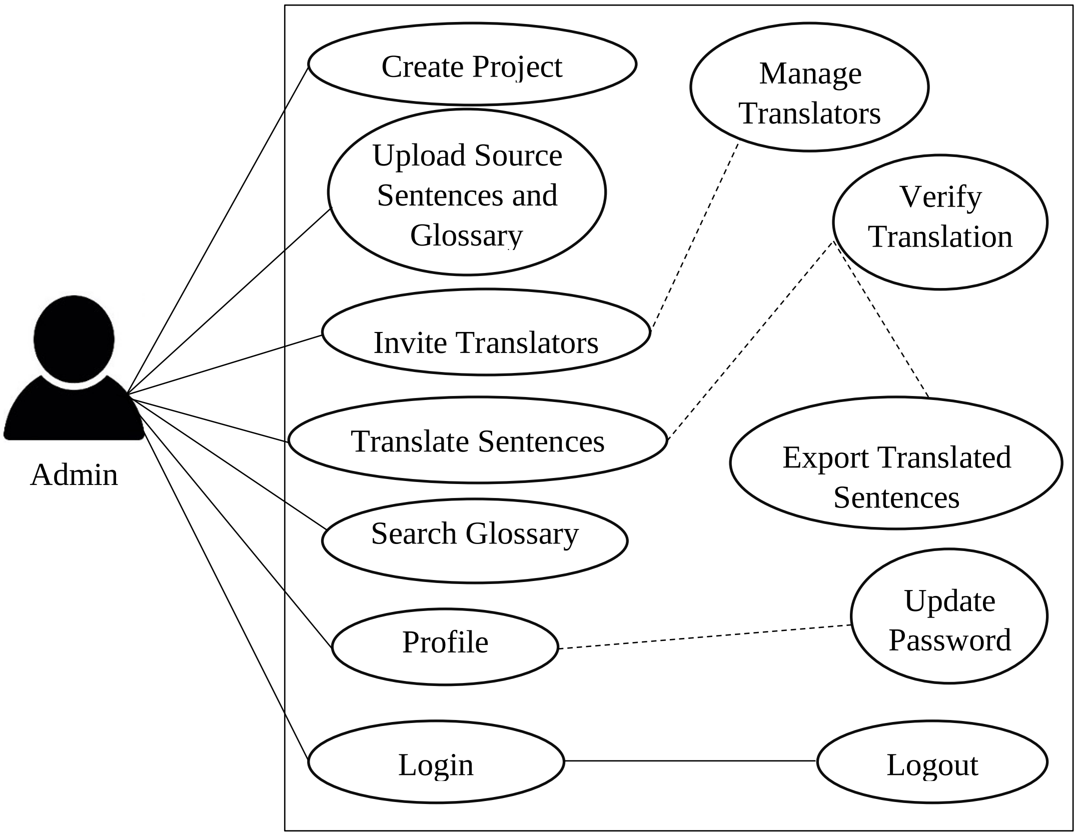


Figure 4.1: Use case model for Admin

In Figure 4.1, we present the use case model for admin. A user as an admin can create a project and maintain the project and its related settings. Maintaining the project includes uploading source sentences which will be translated using the system, uploading glossary which will help the translators by providing suggestions during translations, inviting and managing translators (managing translators refers to change the translators roles either beginner or expert), search glossary item, profile information update such as name and password update, verify translated sentences and export the verified translated sentences as a parallel corpus.

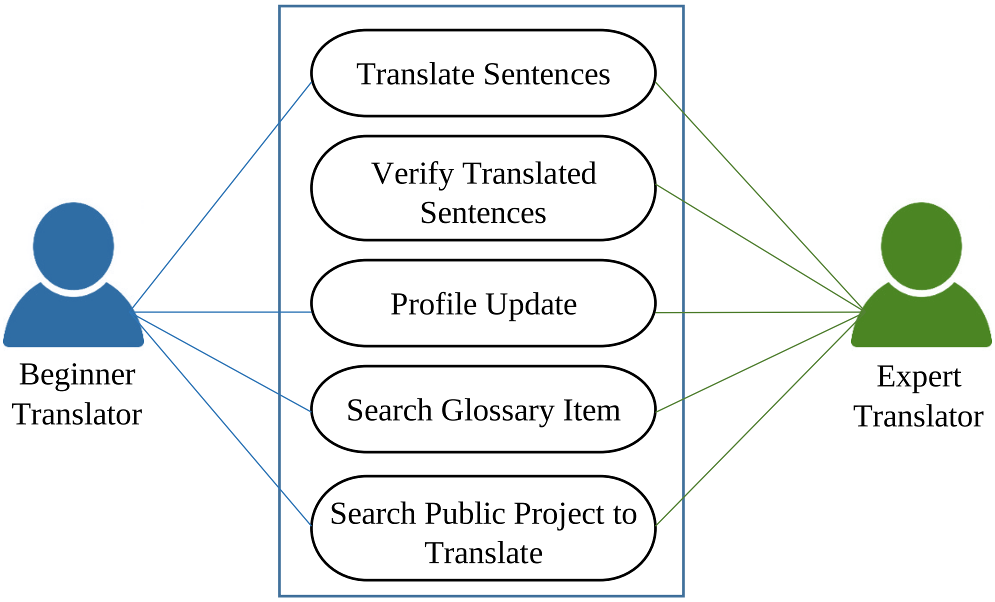


Figure 4.2: Use case model for beginner and expert translators

In Figure 4.2, we represent the user case model for the beginner and expert translators. The main difference between beginner and expert translator is expert translator can verify translated sentences where beginner translator can’t. Otherwise, all the features for both translators are same. Both translators can translate sentences, update profile, searching in glossary items and search for the public project for contributions.

4.2 Entity Relationship Diagram

Entity Relationship Diagram use for describing the relationship of the information or the data of a system. In this section, we will show the ER diagram of database structure of our system. In the database, we twelve tables, each of the tables are connected with each other with at least one column. Figure 4.3 shows the relations among the tables.

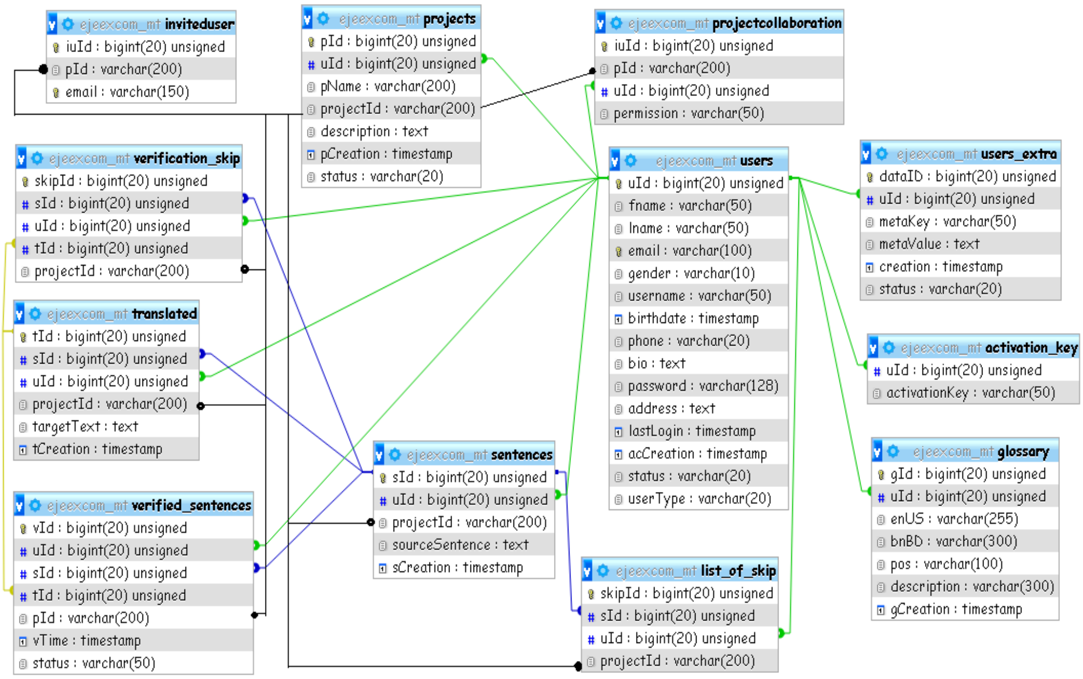


Figure 4.3: ER Diagram of Database structure

4.3 Collaborative Platform Development

In Figure 4.4, we present the system architecture of our parallel corpus development collaborative platform, named as *AmaderCAT*. As a starting point of the process admin uploads sources sentences and glossary files, which are stored in the database. Admin has the role of creating credentials for the translators and any modification of translator's roles are stored in the database. On the translation interface users (i.e., translators) needs to login to the system. While translators start translating the TM database will grow and starts showing the suggestions using fuzzy matching. In our system, we implemented edit-distance for fuzzy matching [9]. The mismatch between the source segment and the TM segment is easy to detect [16] using edit distance. For edit distance method, the distance was taken zero matching each word. Glossary suggestion is one of the most important tools for Computer-Assisted Translation. Our implementations support uploading glossary for any language, which is typically done by project admin. Therefore, if project admin upload glossary and it is present in the system then the system will use it for further suggestions. We used fuzzy approximate string matching for glossary suggestions. These automatic suggestions will help translators to translate the sentences, which our system stores in the database for evaluation and verification. This verification process is included in our system so that expert translator can verify crowd (i.e., non-expert) translators. Upon verification of the translated sentences, they will be stored in the database and TM.

We developed a Translation Memory which will use translated sentences for showing suggestion to the client side and a Glossary suggestion. TMs are a tool for human translators [9]. We builds an advance TM which use translated as its data set. An Edit-Distance Model for the Approximate Matching [9] used for matching Translation Memory. The mismatch between the source segment and the TM source segment is easy to detect [16] using edit distance. For edit distance method, distance were taken zero matching each word. Glossary suggestion is one of the most important tool for Computer Assisted translation. Fuzzy Approximate String matching used for glossary suggestions.

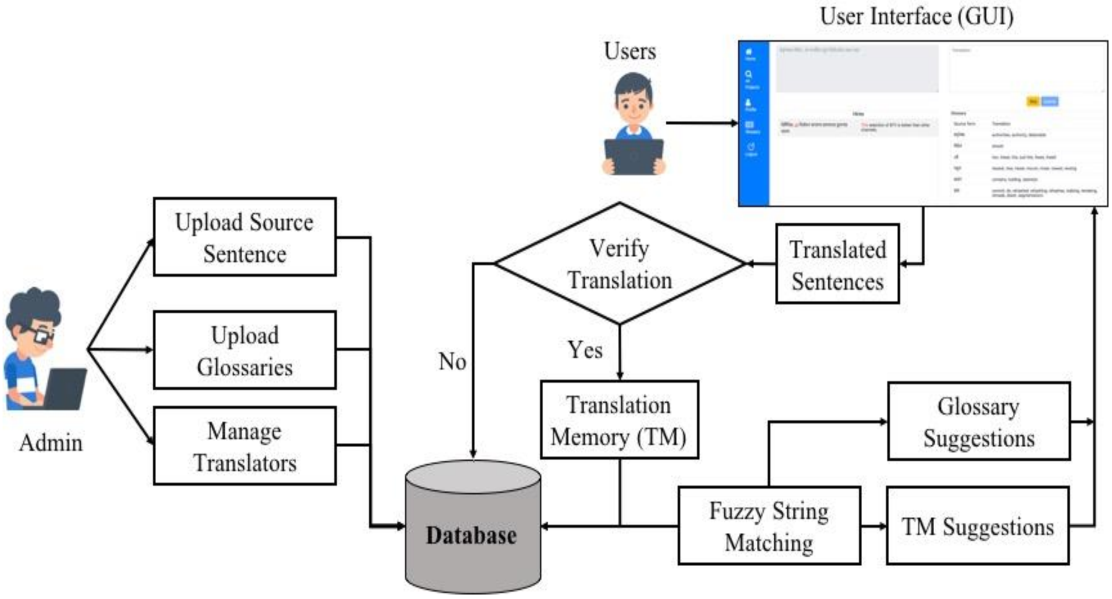


Figure 4.4: System architecture of our parallel corpus development platform.

Additional care needed to be taken into consideration for morphologically rich languages such as Bangla. In Bangla language there are a lot of inflected forms for a single word. It is quite difficult to determine the stem words from inflected words in bangla as it is one of the most morphologically rich languages and it has lots of inflectional and derivational variant forms of a word [8]. In Bangla noun and verb changes, the root word in a different form even the root word may not be extracted from the inflected word. Root words are extracted for getting the meaning of the inflected word and showing the meaning to the translators. For glossary and TM suggestion, we listed more than 200 suffix words. Noun, verb and person suffix were included in the list. We used a rule-based Bangla stemmer for getting root words from inflected words [18]. These root words are used for finding glossary suggestions from glossary items and TM suggestions from TM using fuzzy matching [9].

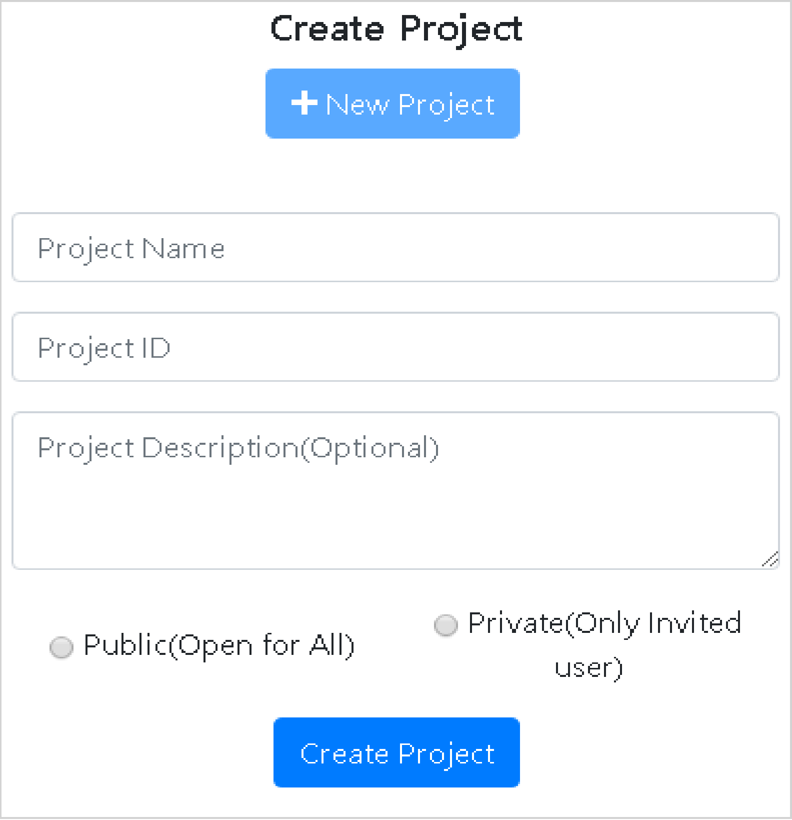


Figure 4.5: Create Project.

From Figure 4.5, we present a screen-shot of the project creation page, which requires a project name, a unique project identity, a project description text box and an option for selecting the type of project. The description text box enables users to describe the purposes and provide a brief description of the project. There are two project types, such as public and private. Public project type enables all the translators who are registered in the system to contribute to the project. Whereas with the private project type only invited translators can contribute, which is managed by the project creator or project admin. The invited translators can register by accepting the email invitation or can register themselves. If the invited person is not registered in the system then the user will be asked to register. The translators are two types such as beginner and expert and these types are not the same for every project. Beginner translators can only translate sentences and the expert translators can translate and verify translated sentences. The system has the functionalities and the invited person if not registered will receive an email for registration. In terms of translation, the translators can translate both public and invited projects, however, in order to contribute to the private project the admin of the private project needs to approve who can contribute to the project.

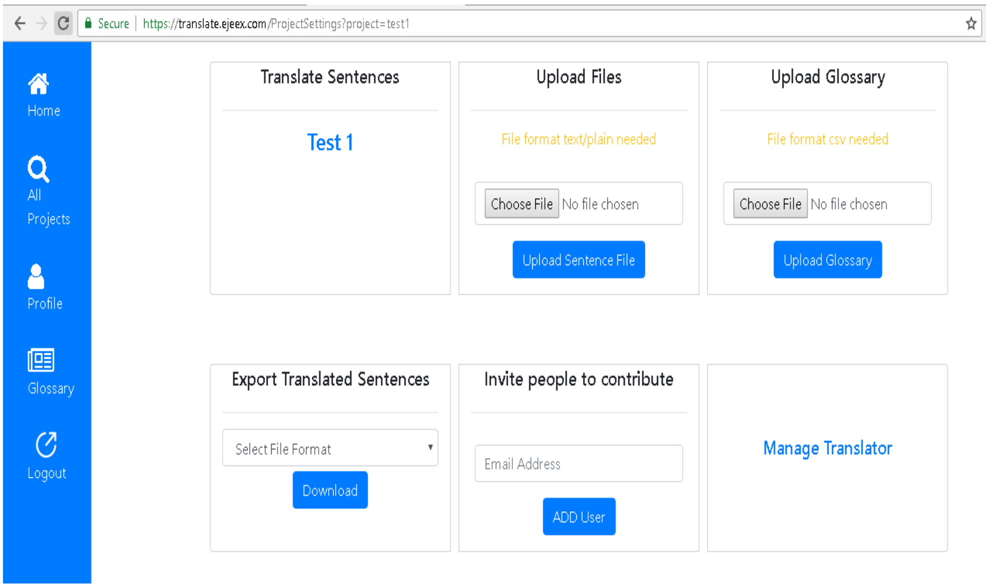
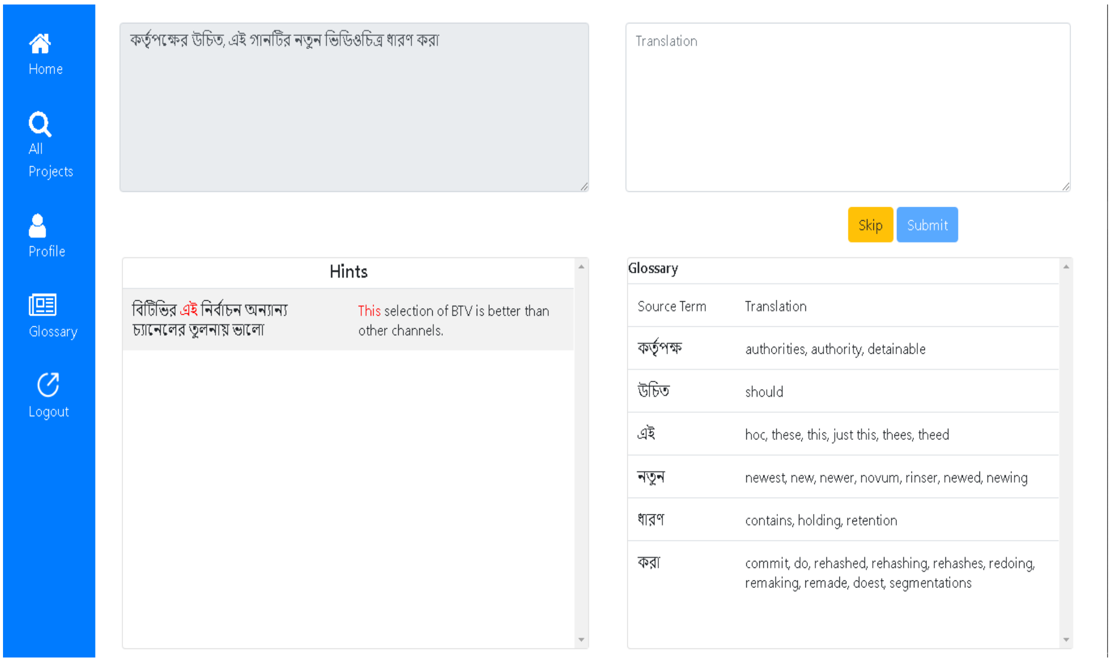


Figure 4.6: Project Settings for the project creator/admin.

We Figure 4.6, we present another screen-shot, which demonstrates how a project needs to be set up. It requires a source sentences file, which needs to be translated and the file format must be in plain text. Project creators can upload glossary files for the project but the glossary files are not mandatory and the file format must be in csv (comma separated) format. If no glossary files present in the project then the system will use system glossary to show glossary suggestion. For the system, we developed a unified glossary by collecting data from different dictionaries, which we use as a system glossary. Currently, we only implemented Bangla-English glossary list as a system glossary. We plan to include more language pairs in our future work.

In Figure 4.7, we present a screen-shot of our system to demonstrate how the translation process works. Upon open the translation interface the system show the sentence to be translated. At the same time, it shows glossary and TM suggestions. Such suggestions can facilitate translation works.



কর্তৃপক্ষের উচিত, এই গানটির নতুন ভিডিওচিত্র ধারণ করা

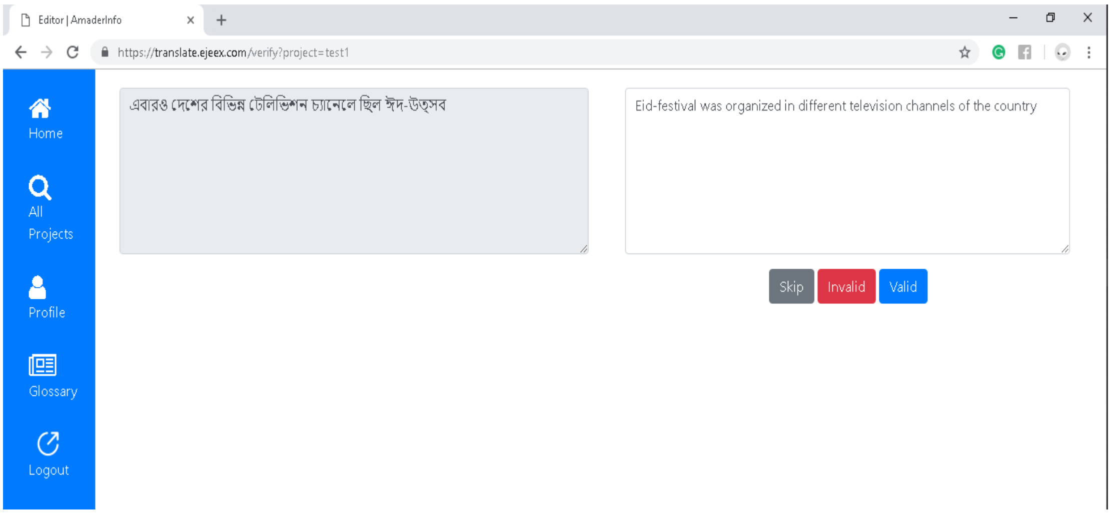
Hints

বিটিভির এই নির্বাচন অন্যান্য চ্যানেলের তুলনায় ভালো	This selection of BTV is better than other channels.

Glossary

Source Term	Translation
কর্তৃপক্ষ	authorities, authority, detainable
উচিত	should
এই	hoc, these, this, just this, thees, theed
নতুন	newest, new, newer, novum, rinser, newed, newing
ধারণ	contains, holding, retention
করা	commit, do, rehashed, rehashing, rehashes, redoing, remaking, remade, doest, segmentations

Figure 4.7: Translation editing interface for the translator.



https://translate.ejeex.com/verify?project=test1

এবারও দেশের বিভিন্ন টেলিভিশন চ্যানেলে ছিল ঈদ-উত্সব

Eid-festival was organized in different television channels of the country

Figure 4.8: Translation verifying interface for the expert translator and project creator

Once a translation process completes by a beginner it can be verified (*Figure 4.8*) by an expert and can approve, which could be the final translated sentences. At the same time, the verified translated sentences will be saved in the TM database. In order to use the translated corpus for a further task such as Machine translation, the project creator can download the translated sentences whenever needed.

CHAPTER 5
DATA COLLECTION

The main motivation in developing *AmaderCAT* system was to develop parallel corpora for Machine Translation. We believe that using our preliminary version of the system one can develop parallel corpus for any language pair such as Bengali to English, English to Bengali, and English to Spanish etc. To translate the corpus using our system we randomly selected 2,000 sentences from Bangla newspapers. For translating those sentences, we assigned few sentences to the students of English department of Daffodil International University to translate in hands. For translating the rest of the sentences, we invited few students to translate those sentences using our system. As a result, we got 1,800 translated sentences out of 2,000 Bengali sentences.

In our parallel corpus, we have ~15,064 Bengali words in 1800 sentences and ~19,102 English words in 1,800 sentences where the maximum number of words for both Bengali and English in a sentence are 12 and 30, respectively. The average number of words in a Bengali sentence is 7.53 words per sentence where an average number of words English sentence is 10.61 words per sentence. In our corpus, we have 6,640 unique Bengali words and 4931 unique English words. While we calculated unique words for Bengali dataset, we ignore the inflection forms.

CHAPTER 6

SYSTEM TESTING AND EVALUATION

6.1. System Testing

Our system needs to be tested properly from end-to-end before publicly available of the system for the end users. There are few basic testing for most of the application before publishing. These are:

- Functionality Testing: check all the links in the application, database connection, forms used for getting or submitting data from the user to application, and cookie testing.
- Usability Testing: especially human-computer interactions and weaknesses are measured during this test.
- Interface Testing: is done by verifying that communication is done properly.
- Compatibility Testing: is performed by using different browsers, OS, mobile, tablet, etc. And the goal is to obtain same performance in all the platform.
- Performance Testing: is use for measuring the behavior under heavy load.
- Security Testing: prevent unwanted access of the user.

Table 6.1: Test case for '*AmaderCAT'* collaborative platform.

Test Case	**Test Input**	**Expected outcome**	**Obtained outcome**	**Pass / fail**	**Tested on**
1. Registration	Empty first name, last name, email, mobile number, password	Show warning to Fill all the required fields	Fields must be filled by data	Pass	15-07-2018
2. Login	Login using various devices such as tablet, pc ,cell phones	Successfully login	Successfully login	Pass	15-07-2018
4. Password	Incorrect password or empty field	Warn the incorrect password or field is empty	Show warning	Pass	15-07-2018

4. Create Project	Fill Project name, id and description	Successfully project created	Successfully project created	Pass	15-07-2018
4. Upload source sentences	A plain/text file with source sentences	Successfully uploaded	Successfully uploaded	Pass	15-07-2018
5. Upload glossary items	A comma-separated/CSV file	Glossary uploaded successfully	Glossary uploaded successfully	Pass	15-07-2018
6. Invite people for contributions	Valid e-mail address	Sending e-mail to the user.	E-mail sent.	Pass	15-07-2018
7. Manage Translators	Changing translator permission	Restrict or Explore translators access	Translator permission changed.	Pass	15-07-2018
8. Translate sentences	A valid/invalid equivalent meaning of input sentences	Next sentence will appear	Next sentence appear	Pass	15-07-8017
9. Verify translations	Mark as valid or Invalid or Skip	Next sentence will appear for verifying	Next sentence appear	Pass	15-07-2018
10. Search glossary item	Type a word or letters in search bar	Showing matching words	Shows matching words	Pass	15-07-2018
11. Profile settings	View profile, Update profile	Show and update profile information	Show and update information successfully	Pass	15-07-2018

6.2 System Evaluation

In order to understand the usability of our system we evaluated using *cognitive walkthrough approach* [19, 23]. It is an approach in which one or more evaluators perform a series of tasks to evaluate the system in terms of understanding the system's learnability for new users. For the experiment, we invited few students from the English department to participate in the evaluation and 7 students have participated. By following the *cognitive walkthrough approach* we designed the following list of questions. Five tasks were given among the students to perform by using our system and asked them to answer the questions using the Likert scale method [2]. The scoring of the Likert scale was calculated between -2 to 2.

1. The application has a user-friendly interface?
2. The application is easy to navigate?
3. The application allows the user to upload files easily?
4. You tried and achieve the right outcome?
5. Correct action available to you to reach your goal?
6. The outcome you expect to achieve comes from the right action?
7. Correct action is performed and the progress is being made towards your intended outcome?

While performing the task we calculated the task completion time. At the same time we compute *task completion rate* and *number of error per task* using the Equations 1 and 2. To understand the effectiveness of our system, we run the same set of experiments using *OmegaT*.

$$Completation\ Rate = \frac{Number\ of\ Task\ Completed}{\text{Total Task}} \times 100\% \qquad (1)$$

$$Number\ of\ Errors\ Per\ Task = \frac{Number\ of\ Error}{\text{Total Task}} \qquad (2)$$

Table 6.2: Evaluation of the system. Task completion rate higher is better, error rate lower is better and average usability score higher is better.

Metric	**AmaderCAT**	**OmegaT**
Task completion time	4.8 min.	4.95 min
Task completion rate	94.29	85.71
Error Rate	0.23	0.34
Average usability score	64	53

In Table 4.1, we present results comparison of the two systems. From the table is clearly visible that *AmaderCAT* gets higher usability score than *OmegaT*. In general out of 7 students 6 students recommended *AmaderCAT* over *OmegaT*. It is needed to mention that the number of users for this evaluation experiment are low, therefore, we plan to run another experiment with more users to prove that our results are statistically significant.

CHAPTER 7
DISCUSSION AND CONCLUSION

7.1 Discussion

The parallel corpus development is a challenging task due to the fact that it is memory intensive, requires a high cognitive load. In order to make the task easier for the translators, it is necessary to provide them with useful tools that can facilitate them in the translation process. Even though there have been publicly available tools, however, there are some limitations in order to use them in a collaborative manner with crowd translators. As a result, we developed *AmaderCAT*, which will significantly help the research community. As this is a very early version of our work, hence, we hope to improve it in the future. In addition, we also plan to make the translated parallel corpus available for the research community.

7.2 Conclusion

In the study, we present a parallel corpus development tool *AmaderCAT,* which open-source and will be made publicly available. The main functionality of the system is that it is collaborative and can be used for crowd translation work. It has great potential for a large-scale parallel corpus development work. In order to check how the system works in terms of usability, we run experiments using *cognitive walkthrough approach*, which proves that *AmaderCAT* has more potential compared to *OmegaT*. We have completed our first set of data collection, which plan to make it public for Bangla Machine translation research.

We believe more functionality can be added to the system to make it more useful, which we plan to do in the future. Our forthcoming work also includes more data collection and developing Bangla Machine translation system.

REFERENCES

[1] Ahmed, S., Rahman, M.O., Pir, S.R., Mottalib, M., Islam, M.S.: A new approach towards the development of english to bangla machine translation system. In: International Conference on Computer Information and Technology (ICCIT). Pp.360–364 (2003)

[2] Allen, I.E., Seaman, C.A.: Likert scales and data analyses. Quality progress40 (7),64–65 (2007)

[3] Asaduzzaman, S., Ali, M.M.: Transfer machine translation-an experience with bangla english machine translation system. In: the Proceedings of the International Conference on Computer and Information Technology (ICCIT), Bangladesh (2003)

[4] Ashrafi, S.S., Kabir, M.H., Anwar, M.M., Noman, A.: English to bangla machine translation system using context-free grammars. International Journal of Computer Science Issues (IJCSI)10(3), 144 (2013)

[5] Brown, P.F., Pietra, V.J.D., Pietra, S.A.D., Mercer, R.L.: The mathematics of statistical machine translation: Parameter estimation. Computational linguistics19 (2), 263–311 (1993)

[6] Chiang, D.: A hierarchical phrase-based model for statistical machine translation. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. pp. 263–270. Association for Computational Linguistics (2005)

[7] Cho, K., Van Merri¨enboer, B., Bahdanau, D., Bengio, Y.: On the proper-ties of neural machine translation: Encoder-decoder approaches. arXiv preprintarXiv:1409.1259 (2014)

[8] Das, S., Mitra, P.: A rule-based approach of stemming for inflectional and derivational words in bengali. In: Students' Technology Symposium (TechSym), 2011IEEE. pp. 134–136. IEEE (2011)

[9] Dobriˇsek, S.,ˇZibert, J., Paveˇsi´c, N., Miheliˇc, F.: An edit-distance model for the approximate matching of timed strings. IEEE Transactions on Pattern Analysis &Machine Intelligence (4), 736–741 (2008)

[10] Escart´ın, C.P.: Design and compilation of a specialized spanish-german parallel corpus. In: LREC. pp. 2199–2206 (2012)

[11] Harshawardhan, R., Augustine, M.S., Soman, K.: Phrase based english–tamil translation system by concept labeling using translation memory. International Journal of Computer Applications20(3), 1–6 (2011)

[12] Hummel, J., Knyphausen, I.: Method and apparatus for processing source information based on source placeable elements (Mar 28 2006), uS Patent 7,020,601

[13] Koehn, P.: Europarl: A parallel corpus for statistical machine translation. In: MTsummit. vol. 5, pp. 79–86 (2005)

[14] Koehn, P.: Statistical machine translation. Cambridge University Press (2009)

[15] Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N.,Cowan, B., Shen, W., Moran, C., Zens, R., et al.: Moses: Open source toolkit for statistical machine translation. In: Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions. pp. 177–180. Association for Computational Linguistics (2007)

[16] Koehn, P., Senellart, J.: Convergence of translation memory and statistical ma-chine translation. In: Proceedings of AMTA Workshop on MT Research and the Translation Industry. pp. 21–31 (2010)

[17] Lagoudaki, E.: Translation memories survey 2006: Users perceptions around tmuse. In: proceedings of the ASLIB International Conference Translating & the Computer. vol. 28, pp. 1–29 (2006)

[18] Mahmud, M.R., Afrin, M., Razzaque, M.A., Miller, E., Iwashige, J.: A rule based bengali stemmer. In: Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on. pp. 2750–2756. IEEE (2014)

[19] Nielsen, J.: Usability inspection methods. In: Conference companion on Human factors in computing systems. pp. 413–414. ACM (1994)

[20] Ruiz Yepes, G., et al.: Parallel corpora in translator education (2011)

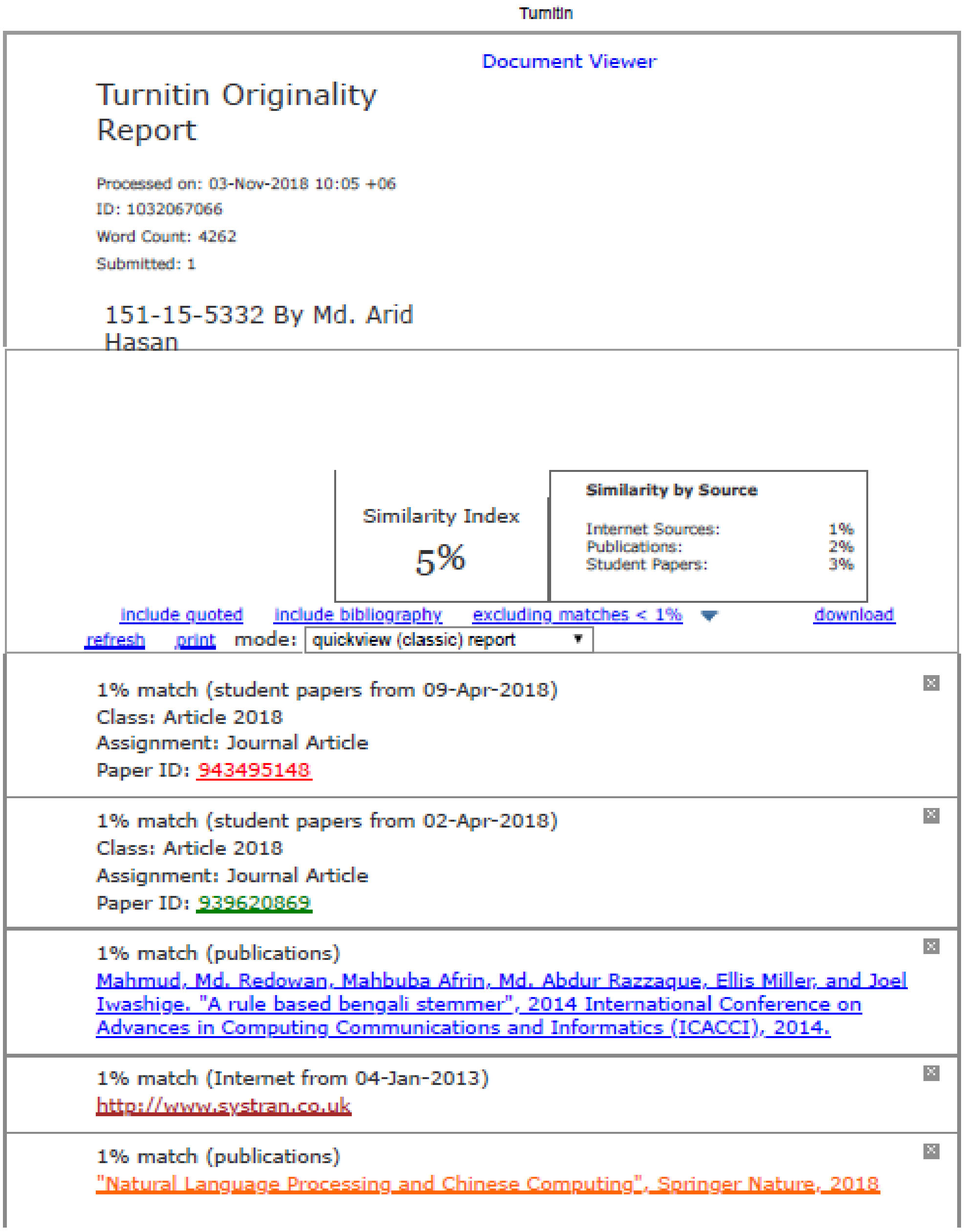
[21] Skadi ̧nˇs, R., Puri ̧nˇs, M., Skadi ̧na, I., Vasil ̧jevs, A.: Evaluation of smt in localization to under-resourced inflected. In: 15th international Conference of the European Association for Machine Translation. pp. 35–40 (2011)

[22] Somers, H.: Translation memory systems. Benjamins Translation Library35, 31–48 (2003)

[23] Wharton, C.: The cognitive walkthrough method: A practitioner's guide. Usability inspection methods (1994)

[24] Zampieri, M., Vela, M.: Quantifying the influence of mt output in the translators' performance: a case study in technical translation. In: Proceedings of the EACL2014 Workshop on Humans and Computer-assisted Translation. pp. 93–98 (2014)

PLAGIARISM REPORT



Turnitin

Document Viewer

Turnitin Originality Report

Processed on: 03-Nov-2018 10:05 +06
ID: 1032067066
Word Count: 4262
Submitted: 1

151-15-5332 By Md. Arid Hasan

Similarity Index
5%

Similarity by Source	
Internet Sources:	1%
Publications:	2%
Student Papers:	3%

mode: quickview (classic) report

1% match (student papers from 09-Apr-2018)
Class: Article 2018
Assignment: Journal Article
Paper ID: 943495148

1% match (student papers from 02-Apr-2018)
Class: Article 2018
Assignment: Journal Article
Paper ID: 939620869

1% match (publications)
Mahmud, Md. Redowan, Mahbuba Afrin, Md. Abdur Razzaque, Ellis Miller, and Joel Iwashige. "A rule based bengali stemmer", 2014 International Conference on Advances in Computing Communications and Informatics (ICACCI), 2014.

1% match (Internet from 04-Jan-2013)
http://www.systran.co.uk

1% match (publications)
"Natural Language Processing and Chinese Computing", Springer Nature, 2018