ALLex: Automated Marking of Language Learning Exercises

[An NLG-based language learning platform with automated question generation and marking]

Kgotso Nkosi
Department of Computer Science
University of Cape Town
Western Cape
South Africa
nkskgo001@myuct.ac.za

Nikhil Gilbert
Department of Computer Science
University of Cape Town
Western Cape
South Africa
glbnik001@myuct.ac.za

Soham Hanuman Singh
Department of Computer Science
University of Cape Town
Western Cape
South Africa
sngsoh004@myuct.ac.za

ABSTRACT

This proposal is a motion to research how statistical techniques can be applied to the creation of a computer assisted language-learning platform (CALL). The proposal discusses the methods, planning and drawbacks associated with the research, which is divided into the processing, generation and classification of language learning questions. The proposal begins by giving a short description of the problem and discusses the importance of a CALL platform for isiZulu. We then discuss our methodology and procedures, any legal and ethical matters, related works, anticipated outcomes and lastly we give a breakdown of the task allocations and plan for the project.

1. PROJECT DESCRIPTION

This project is designed to respond to the lack of automated language generation systems in computer assisted language learning (CALL) for Bantu languages. Manual marking of language learning exercises is cumbersome. It takes time and resources to create and mark unique exercises that are good for teaching a language when done physically (i.e. with writing materials). There are also problems with assessing the difficulty of these tasks, and as such, there is no way to automatically assess whether someone is progressing in a language by being able to answer more and more difficult questions [6]. We want to solve this problem by creating a language learning platform in order to conduct research on using statistical learning techniques to understand, generate and classify language learning exercises. The platform should be able to mark language questions, give users feedback using automated marking, and notify them of how difficult the questions they are answering are.

1.1 Project Significance

CALL is a term for all methods that involve computer aided efforts in teaching human being's natural languages [12]. It can involve any kind of method traditionally used for language learning, such as word completion tasks, or techniques that are more technologically advanced processes that require voice input to check pronunciation and expression. CALL has numerous benefits in situations where teachers or human assisting agents are absent [12].

The number of CALL systems for isiZulu and other Bantu languages are rare, and from our research thus far, none of these systems provide randomized sentence generation that allow for dynamic learning. Platforms that have a set number of questions are generally susceptible to various types of practice effects, and learners cannot reinforce concepts at an acceptable level [9]. Practice effects also correlate a deterioration in performance when new material is presented [9]. It is for this reason that a system that can reinforce grammatical concepts with a large body of vocabulary will prove beneficial in gaining breadth of acquisition of isiZulu.

1.2 Project Issues and Difficulties

Language learning systems for isiZulu are not well grounded in academic literature. This deficit in literature is applicable to all portions of our research due to the fact that Bantu languages are not well covered in statistical approaches to NLP. The availability of a good corpora on which to perform natural language processing is also scarce and this creates difficulties on having adequate infrastructure to create a research platform with.

The challenges with generation for the algorithm we will be using stem from the fact that isiZulu grammar has not been mapped to first order logic. However, we aim to deal with this problem by using other formalisms that can be easily placed into data structures that can be used for statistical processing. Keet and Khumalo [5] show how isiZulu verb rules can be specified as a context free grammar. We aim to use context free grammars of this type and use general patterns to convert them into first order logic.

It is likely that an ontological representation of the isiZulu language, or concept mappings of the language, will be required for this project as well. These resources may be difficult to acquire. Whilst language concept mappings may be obtainable for isiZulu, they would then need to be converted from physically written data into coded data structures so that they may be used for the project.

Logistical issues also exist in that the team will be primarily based in Cape Town (being that we are students at the University of Cape Town). However, the University of Cape Town does not offer isiZulu as a module. This would require a member of the team to travel, or communicate via the internet, in order to conduct the user testing with isiZulu language learning students.

2. PROBLEM STATEMENT

The problem we are addressing is the lack of availability of methods to create CALL systems for isiZulu. We aim to solve this by researching methods that can process an isiZulu corpus, generate natural language questions from this annotated corpus, and classify these questions based on difficulty.

Communication amongst South African people requires advanced methods for a large population to learn languages. This is beneficial to the populous by creating good communication between communities and individuals [1]. Moreover, a lack of trained human resources for teaching in South Africa makes it difficult for learners that are interested in learning languages to make progress without physical teaching support and adequate practice questions [11]. The problem with learning and being assessed by a static system is that it does not provide a variety of examples to improve vocabulary and create new challenges for improvement [6].

2.1 Aims and Research Questions

Our paper aims to develop new methods for CALL platforms, in order to make them more effective at generating random questions that do not have repetition or patterns that can be easily learned. This approach will ensure that learners are more challenged in various aspects of language learning, such as becoming more familiar with vocabulary in similar grammatical circumstances. Additionally, our paper aims to expand the body of literature available for NLP in Bantu languages, and hopefully inspire further study for future researchers.

Below are the research questions this project will be used to explore for each team member:

- **Kgotso:** Can the EMMA be extended to a fitness function of the local (1+1) EA to develop strong morphological analyzers?
- **Nikhil:** How do decision trees (PLTAG) need to be constructed from isiZulu grammar in order to solve a Markov Decision Process, to generate natural language sentences from a morphologically annotated corpus?
- **Soham:** Does using evaluation techniques for question difficulty and similarity to compose CALL assessments, in accordance with Bloom's taxonomy, improve the satisfaction and performance of language learning students?

3. PROCEDURES AND METHODS

3.1 Natural Language Processing

Iterative prototyping will be used to develop the parameters of the evolutionary algorithm. It will be used in conjunction with existing morphological analysis techniques to comply with isiZulu morphology. The result of the algorithm will be an isiZulu POS-tagger. This initial stage of the pipeline is the processing of the corpus. The resulting annotated corpus will then be passed to the natural language generator.

The algorithms performance will be evaluated statistically on a results basis. The proposed evaluation metric will leverage graph theory by generating a bipartite graph of the proposed answer key and the actual answer key, and finding an optimal matching of the two.

3.2 Natural Language Generation

The problem of natural language generation will be solved using the annotated corpus that will be passed from the morphological processing component. The goal in this section is to create a lexicon out of the corpus for the purpose of randomly generated sentences that use different vocabulary but follow the same grammatical structure. The task is split into two phases: preparation and generation.

3.2.1 Preparation

Using the technique described by Nathan and Ray [7], a subset of isiZulu grammar using First-order logic will be constructed so that we may create what they refer to as PTAG trees. This declares the exact format of a particular aspect of the construction of a particular sentence in isiZulu. Then, using the words in the annotated corpus, we will create what their research referred to as a PLTAG tree (see figure 1) [7]. This is essentially a PTAG tree that has been populated by words from the corpus [7]. In the paper by Nathan and Ray, this is referred to as a lexicon [7]. We will aim to construct enough grammars so that a large enough variety of sentences can be formulated for the classification model to assess their difficulty, and will allow for learning tasks with an adequate amount of grammar variance.

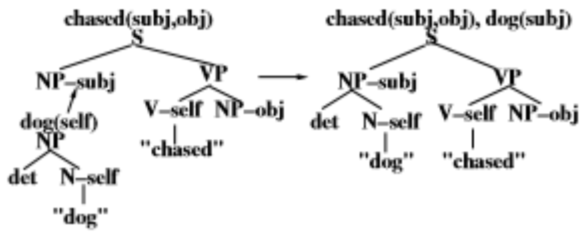


Figure 1: An example of a PLTAG tree for an English language sentence [7].

3.2.2 Generation

Once the tree is generated, we can use the Markov Decision Process model to traverse the tree and generate a sentence. The MDP works by starting at a valid location of the tree and continues to use adjoins, adding words to the sentence until the communicative goal is complete [7]. See figure 2 for the equation the policy used by the tree to solve the MDP. The MDP needs to be planned such that each element it visits in the tree will eventually form a correctly structured meaningful sentence once the traversal is complete [7]. The MDP is designed such that we use only the parts of the tree that achieves the communicative goal, hence not every section of the tree needs to be traversed [7]. To ensure variance in the sentence the MDP solution equation allows for increasing rewards for sections of the tree that have not been visited, whilst still accounting for the communicative goal [7].

$$P(s,a) = Q(s,a) + c\sqrt{\frac{lnN(s)}{N(s,a)}}$$

Figure 2: The equation of the policy used by the tree to solve the MDP [7].

3.3 Difficulty and Similarity Evaluation

3.3.1 Input

The Natural Language Generation component will pass on a list of possible questions for the assessment to the Difficulty and Similarity Evaluation component as input. The number of questions that are given as input are dependent of the number of questions required for the assessment (i.e. the more questions required for an assessment, the more possible questions are given as input to this component of the system). There will always be more questions given as input to this component as not all questions will be selected during the exercise composition stage.

3.3.2 Similarity Processing

It is important to check whether or not the questions provided as input do not contain any questions that are very similar, or in fact, identical to one another. Before proceeding to difficulty evaluation, questions will first be checked for similarity. All questions that are deemed identical to another question will be removed. A similarity metric will determine the degree of similarity of a question to another question and in the case where a question is extremely similar and the system can afford to lose more questions without falling below the minimum required for the exercise, then the extremely similar questions will be removed as well.

3.3.3 Difficulty Evaluation and Assessment Composition

After having received the set of questions with identical and possibly similar questions removed, they then need to have their difficulty evaluated. Each question will have their difficulty evaluated, with different evaluation methodologies being applied to the relative different question types. After the difficulty of the questions have all been evaluated, an assessment will be created using a subset of these questions. Selection of questions for the assessment will be on the basis of the levels of Bloom's taxonomy [3]. With particular emphasis on majority of the questions being of the "Knowledge", "Comprehension" and "Application" levels of Bloom's taxonomy and a select few questions, if any, being in the "Analysis", "Synthesis" and "Evaluation" levels [3]. This then results in the completion of the final assessment to be output by this component as the final product of the entire system.

3.4 Testing

After the assessment composition modules of the system have been completed (steps 3.1 - 3.3 above) and are working correctly, the assessments generated by the system will be tested by potential users of the system. This will involve a control group of isiZulu students completing assessments whereby a random subset of questions is taken from those generated by 3.2. The experimental group will be isiZulu students who will have to complete assessments whereby the assessments have first undergone difficulty and similarity evaluation and have been composed in accordance with Bloom's taxonomy. A statistical analysis will be done on the results of the two groups to determine if there is a statistically significant difference in the results of the two groups and whether or not the difficulty evaluation, similarity evaluation and composition w.r.t. Bloom's taxonomy has been beneficial for both student experience and their grade obtained.

4. ETHICAL, PROFESSIONAL AND LEGAL ISSUES

4.1 Testing

Ethical clearance will be required from the UCT Human Research Ethics Committee so that the difficulty evaluation, similarity and exercise composition component (section 3.3) may be tested on potential users. This clearance will be obtained by the submission of an ethical admittance form to the committee to request permission so that sessions may be held with potential users to complete assessments generated by the software and then give feedback as to their opinion on the difficulty level of the assessments. Test subjects will be given a complete description of the project and what their involvement in the testing phase entails prior to them performing any tasks or being interviewed. We will also request permission from the test subjects to question and report on their results from the assessments as well as to use any feedback that they may provide to us.

4.2 Data

The software will not collect any personal information. Its purpose is merely to present students with questions and then mark these questions and return their result. All administrative handling of data (such as the student's name or student number) will be handled by the application using the software (i.e. Vula). The application using the software will be the only entity with permission to request the results of an assessment taken. It then may choose who has permission to view such information (i.e. the student and lecturers etc.).

5 RELATED WORK

The are a variety of statistical approaches to natural language generation. We opted to use the Markov Decision Process as the sentences we needed to generate are not based on any prior user input, nor do they require context for a conversation. For this reason, stochastic processes like the one used by Oh and Rudnicky [8], which account for prior user input to give contextual responses, are largely unnecessary and computationally expensive.

Other statistical literature reviewed included hierarchical reinforcement, where the objective of the generator is more complex and is required to describe things of a spatiotemporal nature. The paper by Dethlefs and Cuayahuitl [2] used hierarchical reinforcement in a virtual environment which had multiple changing variables.

With respect to natural language generation techniques, the primary paper whose research we will be adapting is that of Nathan and Ray [7]. This component of the system would be an adaptation of their STRUCT algorithm [7] and will make use of all the techniques that they use to generate language from an annotated corpus. The aim of this will be to examine the effectiveness of STRUCT in generating a variety of grammatically accurate sentences.

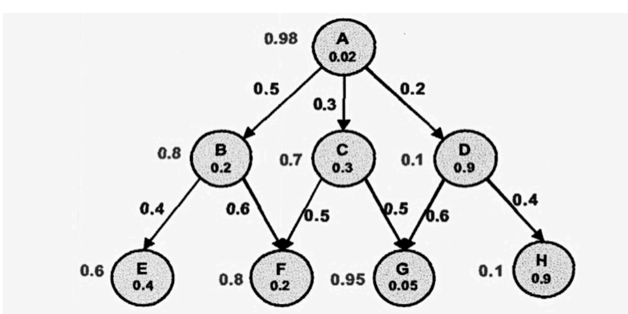


Figure 3: A Topic Dependency Graph with root node A [4].

With respect to the difficulty and similarity evaluation of questions, there already has been a variety of work done. Methods have been developed which employ a variety of tactics, with some using an ontology and others using heuristics or question similarity for the classification process. Khan et al. [4] used an ontology to assess difficulty via two axioms: coverage and diversity. They constructed a Topic Dependency Graph (see figure 3 for an example TDG) for a question and then defined formulas which use the nodes of the TDG as parameters to evaluate diversity and coverage [4]. These two axioms then contributed directly to the difficulty of a question [4].

Ramesh & Sasikumar [10] used an ontology to assess the similarity between two questions. They create subset ontologies (see figure 4 for an example of this subset ontology) for two questions which are being compared to one another. These subset ontologies are then weighted and checked for overlapping [10]. The weighting varies from level to level so that there are degrees of similarity between the two subset ontologies [10].

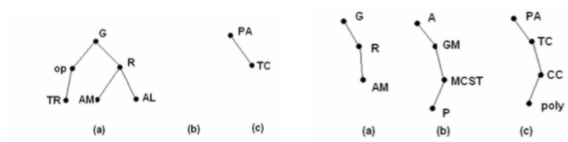


Figure 4: Subset ontologies and their nodes for two separate questions [10].

6 ANTICIPATED OUTCOMES

6.1 System Features

The software artefact to be produce is, among other things, a natural language processing tool for isiZulu. The analyzer should be able to identify the morphosyntactic structure of an isiZulu word by using a machine learning technique (which is a local (1+1) EA) to generate possible answer keys for a labeled isiZulu word list.

Statistical techniques should provide enough grounding for the development of the component which can take the annotated corpus from the natural language processing component, and formulate sentences that can be turned into language learning questions. It will use the PTAG tree defined by Nathan and Ray [7] and the Markov Decision Process that they have modelled. From here, the MDP model will be used to generate a sentence.

Lastly, the difficulty and similarity evaluation component is one which will receive the list of annotated questions and then evaluate the difficulty and similarity of these questions so that it can compose a balanced assessment with respect to Bloom's taxonomy. It will assess the difficulty by two axioms: coverage and diversity [4].

6.2 Design Challenges

We have identified the following as key design challenges for this project:

- Natural Language Processing may be problematic as isiZulu is an agglutinating language which might make the identification of morphemes as well as appropriate lemmatization an issue.

- By using the PTAG tree [7] to formulate sentence, an issue may arise whereby certain isiZulu grammar may not be convertible into a logical formalism which can be used in order to build a tree.

- For difficulty and similarity evaluation, finding an ontological representation of the language or language concept mapping sufficient for this process may be a challenge for isiZulu.

6.3 Expected Impact

We hope that by creating a system of this nature for a language like isiZulu, a member of the Nguni language group, then many more systems similar to this will be developed in the future so that these languages build on their technological resources. We also hope that our final tool will be of use to language learning students who may use the tool to improve their language learning experience. If we are able to identify a statistical correlation between student satisfaction and grade improvement w.r.t. the use of CALL exercises, then we hope that this will encourage teachers and lecturers to adopt CALL exercises as a means of assessment. This will save teachers large amounts of time which would otherwise be spent manually marking and setting assessments.

6.4 Key Success Factors

This automatic morphological analyzer will be evaluated quantitatively. Statistical analysis will be run on the results data to find the recall and precision of the answer keys, and an F-measure.

The NLG component of the project will be deemed successful based on two factors. The first is the successful generation of a valid isiZulu sentence. This is the most basic functionality of the system, as it forms the basis from which the system generates questions. The second is the variety of sentences that can be created using the natural language generator. The generator is meant to form the grounding for a practically implemented CALL platform. This means that the variety of sentences that one can generate should be based on more than one grammar model. For the scope of this project we will consider at least four different grammatical representations an adequate variety.

Success of the difficulty and similarity evaluation and exercise composition component will be judged mainly on the feedback received by potential users who will complete sample exercises made by the system, as well as by a statistical analysis of their mark obtained by completing the assessment against those who completed randomly composed assessments. If students are of the opinion that the assessments are a fair test of their knowledge of the language being taught, and the statistical analysis shows a significant improvement in the learning of the experimental group compared to the control group, then the component can be deemed successful.

7 PROJECT PLAN AND WORK ALLOCATION

7.1 Risk Matrix

We have identified a number of risks associated with this project. These can be seen in Appendix A. The risks that we have identified vary from having a low impact or probability to a high impact or probability. On the whole, the project does contain a relatively significant amount of risk given the lack of available resources for the isiZulu language.

7.2 Timeline

The project period is from the 6th of April, 2017 until the 11th of October 2017. There is a Gantt chart which reflects this timeline (see Appendix B) as well as a Tasks and Milestones table (see Appendix C), both of which indicate the deliverables and progression plan for the project throughout the duration of the development period.

7.3 Required Resources

From a technological perspective, we require computers that are capable of the development of the system. This requires that the computers have the necessary programming languages and libraries, as well as IDEs installed. In order to create a system that can generate and process natural language, we will need an annotated isiZulu corpus and an isiZulu expert to act as our oracle and correct the automatic language analysis and construction. For the difficulty and similarity evaluation component, an isiZulu ontology or isiZulu language concept mapping is required. Lastly, for the user testing stage, isiZulu language learning students are required to complete exercises generated by the system as well as to answer interviews regarding the exercises.

7.4 Deliverables

The CALL program which creates Zulu language learning assessments is the main deliverable for this project. This includes all NLU, NLG and assessment composition by question difficulty components which have been discussed in the previous sections. The other deliverables can be seen in the tasks and milestones table (Appendix C) and include the following (in no particular order):

- a literature review,
- a project proposal (this document),
- a project proposal presentation,
- an initial software feasibility demonstration,
- the final project paper and;
- a final project demonstration.

7.5 Milestones

The Gantt chart (see Appendix B) and the Tasks and Milestones table (see Appendix C) specify the milestones for this project. These include not only project deliverables but as well as targets set by ourselves to ensure development proceeds at a sustainable and sufficient rate. Key milestones include being prepared for the initial software feasibility demonstration (between the 14th and 18th of August) as well as the final code submission for the project due on the 2nd of October, 2017.

7.6 Work Allocation

Care has been taken to modularize the nature of the system so that work can be allocated amongst team members easily. Thus the development of the system has a pipeline nature. Kgotso Nkosi has the responsibility of focusing on the Natural Language Processing component of the system which provides the grammar needed by Nikhil Gilbert, who will be doing the Natural Language Generation (NLG) component (which creates the actual questions for the assessment). Soham Singh will then take the list of questions from the NLG component and perform difficulty and similarity evaluation on each question so that his component may compose an assessment with respect to Bloom's taxonomy. Whilst the systems development has been modularized, all members will work collaboratively during the iterative design process.

REFERENCES

[1] Bak, T.H. and Mehmedbegovic, D., 2017. Healthy linguistic diet: the value of linguistic diversity and language learning across the lifespan.

[2] Dethlefs, N. and Cuayáhuitl, H. 2014. Hierarchical reinforcement learning for situated natural language generation. *Natural Language Engineering*. 21, 03 (2014), 391-435.

[3] Hasan Uğur, Petru-Madalin Constantinescu, and Michael J. Stevens. 2015. Self-Awareness and Personal Growth: Theory and Application of Bloom's Taxonomy. *Eurasian Journal of Educational Research* 15, 60 (January 2015), 89–110. DOI:http://dx.doi.org/10.14689/ejer.2015.60.6

[4] J.i. Khan, M. Hardas, and Yongbin Ma. A Study of Problem Difficulty Evaluation for Semantic Network Ontology Based Intelligent Courseware Sharing.*The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)* (2005). DOI:http://dx.doi.org/10.1109/wi.2005.15

[5] Keet, C.M. and Khumalo, L., 2016. Grammar rules for the isiZulu complex verb. *arXiv preprint arXiv:1612.06581*.

[6] Lai, C.C. and Kritsonis, W.A., 2006. The advantages and disadvantages of computer technology in second language acquisition. *Online Submission*, 3(1).

[7] Nathan, M. and Ray, S. 2014. A Decision Theoretic Approach to Natural Language Generation. *Case Western Reserve University.*

[8] Oh, A. H. and Rudnicky, A. I. 2002. Stochastic natural language generation for spoken dialog systems. *Computer Speech & Language*. 16, 3-4 (2002), 387-407

[9] Pashler, H. and Baylis, G.C., 1991. Procedural learning: I. Locus of practice effects in speeded choice tasks. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 17(1), p.20.

[10] R. Ramesh and M. Sasikumar. 2010. Use of ontology in addressing the issues of question similarity in distributed question bank.*Proceedings of the International Conference and Workshop on Emerging Trends in Technology - ICWET '10* (2010). DOI:http://dx.doi.org/10.1145/1741906.1742068

[11] Spaull, N., 2013. South Africa's education crisis: The quality of education in South Africa 1994-2011. *Johannesburg: Centre for Development and Enterprise*.

[12] Warschauer, M. and Healey, D., 1998. Computers and language learning: An overview. *Language teaching*, 31(02), pp.57-71.

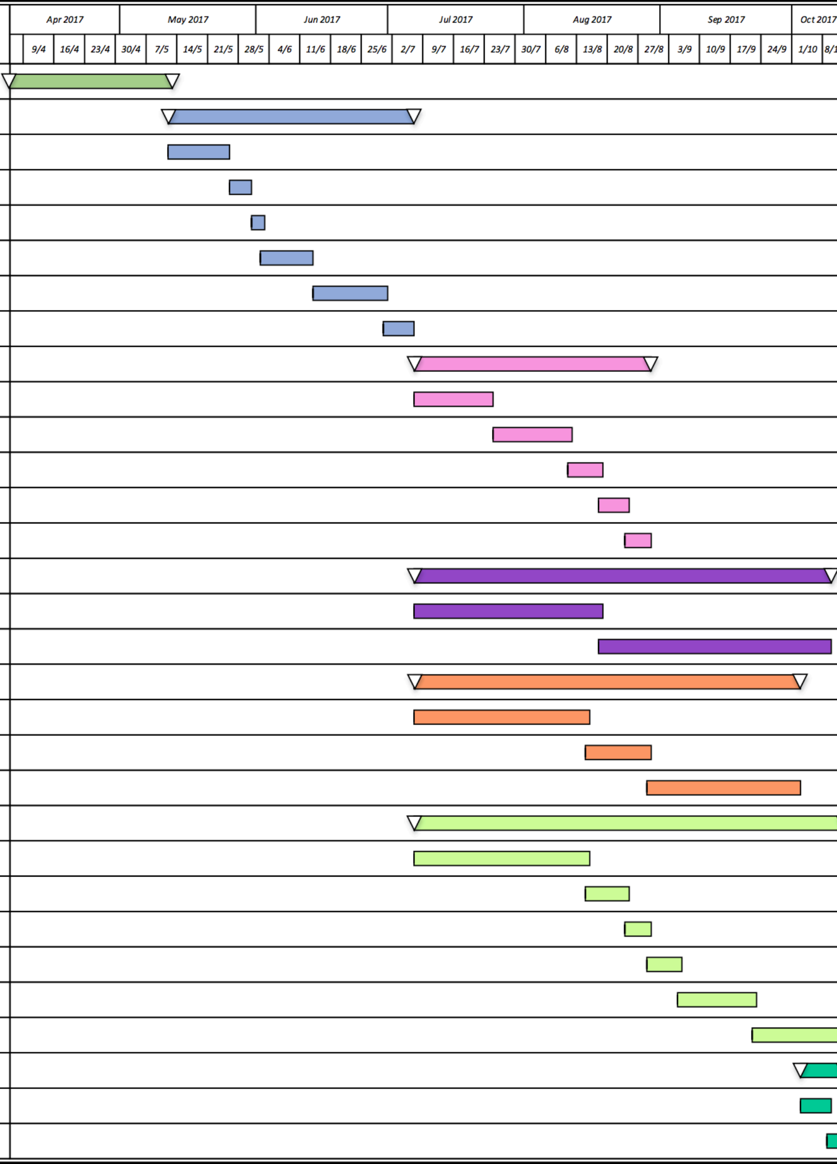
APPENDIX

A. RISK MATRIX

Risk	Impact	Probability	Mitigation
A teammate is unable to complete their part of the system in the required time frame	High	Medium	1. Modularize components so that dummy input can be used as a replacement of the fallen teammates component to test the entire system so that it remains unaffected
Logistical problems to perform isiZulu user testing and interviews	Low	Low	1. Hold Skype sessions with potential users for the testing process and send them the assessment within the session and restrict the time they can work on the exercise 2. Interview done afterwards over Skype as well
Insufficient isiZulu-learning test subjects for user testing	Medium	Medium	1. Perform statistical analysis with smaller sample and make explicitly clear the potential for the sample size to be unrepresentative of the population
Lack of an isiZulu corpus for Natural Language Processing	High	Medium	1. Use the Ukwabelana resource
Lack of an isiZulu Ontology for difficulty and similarity evaluation	Medium	High	1. Use a concept mapping to determine prerequisite concepts and concept coverage and diversity
Scope of the project is too large for the given time period	High	Medium	1. Restrict the scope of the project by minimizing functionality to the bare essentials of the system 2. Add extra components once fundamental functionality is complete
Development takes longer than anticipated	High	High	1. Follow timeline religiously 2. Seek assistance in advance
Certain parts of isiZulu grammar incompatible with First Order Logic	Medium	Medium	1. Only choose grammars that map to First Order Logic 2. Do prior research on grammars to ensure they have grounded formalisms

B. Gantt Chart

ID	Task Name	Start	Finish	Duration
1	**Literature Review**	4/6/2017	5/12/2017	27d
2	**Project Proposal**	5/12/2017	7/6/2017	40d
3	Project Proposal First Draft	5/12/2017	5/25/2017	10d
4	Project Proposal Second Draft	5/26/2017	5/30/2017	3d
5	Project Proposal Hand-in	5/31/2017	6/2/2017	3d
6	Project Proposal Presentation	6/2/2017	6/13/2017	8d
7	Revised Project Proposal	6/14/2017	6/30/2017	13d
8	Project Web Presence Proposal	6/30/2017	7/6/2017	5d
9	**Development**	7/7/2017	8/29/2017	38d
10	Throwaway Prototype and Brainstorming	7/7/2017	7/24/2017	12d
11	Prototype Iteration 1	7/25/2017	8/11/2017	14d
12	Prototype Iteration 2	8/11/2017	8/18/2017	6d
13	Prototype Iteration 3	8/18/2017	8/24/2017	5d
14	Testing	8/24/2017	8/29/2017	4d
15	**Demonstrations**	7/7/2017	10/9/2017	67d
16	Initial Software Feasibility Demonstration	7/7/2017	8/18/2017	31d
17	Final Project Demonstration	8/18/2017	10/9/2017	37d
18	**Code Submissions**	7/7/2017	10/2/2017	62d
19	First Implementation	7/7/2017	8/15/2017	28d
20	Final Implementation	8/15/2017	8/29/2017	11d
21	Project Code Final Submission	8/29/2017	10/2/2017	25d
22	**Report**	7/7/2017	10/23/2017	77d
23	First Experiment/Performance Test + Writeup	7/7/2017	8/15/2017	28d
24	Final Experiment/Performance Test + Writeup	8/15/2017	8/24/2017	8d
25	Final Implementation and Testing	8/24/2017	8/29/2017	4d
26	Outline of Complete Paper	8/29/2017	9/5/2017	6d
27	Project Paper Final Submission	9/5/2017	9/22/2017	14d
28	Reflection Paper	9/22/2017	10/23/2017	22d
29	**Media**	10/3/2017	10/12/2017	8d
30	Poster	10/3/2017	10/9/2017	5d
31	Web Page	10/9/2017	10/12/2017	4d



C. Tasks and Milestones Table

Tasks and Milestones		
Task/Milestone	Start Date	End Date
Literature Review	**06/04/17**	**12/05/17**
Project Proposal	**12/05/17**	**06/07/17**
Project Proposal First Draft	12/05/17	25/05/17
Project Proposal Second Draft	26/05/17	30/05/17
Project Proposal Hand-in	31/05/17	02/06/17
Project Proposal Presentation	02/06/17	13/06/17
Revised Project Proposal	14/06/17	30/06/17
Project Web Presence Proposal	30/07/17	06/07/17
Development	**07/07/17**	**29/08/17**
Throwaway Prototype and Brainstorming	07/07/17	24/07/17
Prototype Iteration 1	25/07/17	11/08/17
Prototype Iteration 2	11/08/17	18/08/17
Prototype Iteration 3	18/08/17	24/08/17
Testing	24/08/17	29/08/17
Demonstrations	**07/07/17**	**09/10/17**
Initial Software Feasibility Demonstration	07/07/17	18/08/17
Final Project Demonstration	18/08/17	09/10/17
Code Submissions	**07/07/17**	**02/10/17**
First Implementation	07/07/17	15/08/17
Final Implementation	15/08/17	29/08/17
Project Code Final Submission	29/08/17	02/10/17
Report	**07/07/17**	**23/10/17**
First Experiment/Performance Test + Write-up	07/07/17	15/08/17
Final Experiment/Performance Test + Write-up	15/08/17	24/08/17
Final Implementation and Testing	24/08/17	29/08/17
Outline of Complete Paper	29/08/17	05/09/17
Project Paper Final Submission	05/09/17	22/09/17
Reflection Paper	22/09/17	23/10/17
Media	**03/10/17**	**12/10/17**
Poster	03/10/17	09/10/17
Web Page	09/10/17	12/10/17