IST-Africa 2020 Conference Proceedings
Miriam Cunningham and Paul Cunningham (Eds)
IST-Africa Institute and IIMC, 2020
ISBN: 978-1-905824-65-6

Question Answering using Automatically Generated Semantic Networks – the case of Swahili Questions

Barack WANJAWA[1], Lawrence MUCHEMI[2]
[1, 2]*University of Nairobi School of Computing, Box 30197, Nairobi, 00100, Kenya*
[1]*Tel: +254 722 614787, Email: wanjawawb@gmail.com*
[2]*Tel: +254 722 234738, Email: lmuchemi@uonbi.ac.ke*

Abstract: Question Answering systems are important in natural language processing applications such as web search and information extraction. Various gold standard curated datasets exist for QA mostly in English. QAs are solved using syntactic, semantic or even learning systems. Low resource languages, such as Swahili, have not been easy to process due to few resources for training, testing and learning. By recognizing that Swahili generally follows subject-verb-object format, which is also the general structure of semantic networks, we use this fact to create a semantic network. Scarcity of public domain corpus has meant that we use Swahili frequently asked questions from public websites, and school texts for Swahili language. We create a semantic network and subject it to QA tasks. The results show good accuracy of about 80% for any typical set of one fact comprehension questions. Challenges such as scarcity of corpora and lack of gold standard test sets remain.

Keywords: Question Answering, Semantic Networks, Swahili, Low resource languages.

1. Introduction

A machine learning system for question answering (QA) should understand both the question part and the answer part so that it can get a best match of these two parts. It is therefore both an information retrieval and information extraction task. There is no restriction on the complexity of the natural language query, since the user can easily query without adherence to expectations such as syntax, well-formed language structure or even the rules of the language itself. QA systems usually have three modules, being the classification of the question, processing of documents and extraction of answer.

A question can be defined as an enquiry that requires unambiguous answerability by humans from some point of reasoning [1]. Such a definition has led to generation of several QA test sets to try to meet this requirement e.g. the MCTest [2] that has stories for comprehension, followed by questions and answers. However, there are many other QA test sets that may be domain specific or that target various aspects of NLP. Some QA test sets include SQuAD [3], the Stanford QA Dataset of 100,000 questions generated by annotators based on Wikipedia articles that can only be answered by understanding the relevant text.

The difficulty with QA remains their performance of machine learning algorithms compared to human gold standards. On a test conducted with the SQuAD QA comprehension dataset, a logistics regression (LR) model could only achieve 51% F1 score, compared to humans who achieved 86.8% [3]. Later results have improved the F1 score to 70.3% [4]. QA systems may be multiple choice based or open ended. The QA system may have questions with answers or there could as well be no answer at all based on the available evidence. All these are challenging aspects for QA processing systems.

There are many other ways of classifying QA systems. They can be classified two broad types – based on reasoning over the natural language (NL) or just NL information retrieval (IR) [5]. However, [6] proposes eight different classification methods based on aspects such as domain, question type, input query analysis, data sources consulted, data source characteristics, representation types, retrieval techniques and answer format. Still others classify QA as IR or NLP [7] or even purely on approach, hence NLP and IR or NLU and reasoning [8]. Still other researchers suggest classification as rule-based, web based, restricted domains and IR based [9]. Other classification methods exist.

QA tasks aim to test various features, which are akin to human reasoning over QA systems e.g. single supporting fact questions (one, two, three or even more), argument relations, yes/no questions, counting and lists, negation and indefinites, coreference, conjunctions etc. [1]. An ideal QA system should be robust enough to address all these features in a generic manner, not on a task-by-task tuning. Such ideal systems are impractical.

Various methods of generating answers in a QA system include simple n-grams, which can be baselines, Long short-term memory (LSTMs), Recurrent Neural Networks (RNNs), memory neural networks (MemNNs) or Support vector machines (SVMs). Pre-processing tasks such as role labelling and coreference resolution are usually done through training based on other external knowledge bases, before the documents are processed in the QA pipeline. Approaches to solving the QA problem also differ, from just considering the semantic structures or using cognitive approaches that try to model the process that humans would follow to answer a question (understand, plan, execute, determine) [5]

Deep learning (DL) systems have found use in solving different Artificial Intelligence (AI) problems by first learning through training on available training data, then making predictions based on new data, or test data. Graph Convolution Neural Networks (GCNs) [10], which is a variant of Convolution Neural Networks (CNNs) have been found applicability in QA tasks with good results. [11] mentions one shortcoming of QA algorithms as not being able to reason to multi-document level.

Resource rich languages have many QA corpora available especially on the web, usually as open source, hence have been widely explored by researchers. On the other hand, resource scarce languages, such as Swahili suffer the disadvantage of not having much corpora for training or even NL pre-processing tools. Twelve different metrics are used to assess how resource rich a language is, including processing tools, speech processing, translation, Optical Character Recognition (OCR) and web resources [12,13,14]. Any language scoring below 10/20 on this scale, such as Swahili, is termed as under-resourced. This is despite Swahili being an important language in Africa, spoken by over 40 million people mainly in East and Central Africa [15]. The most popular text corpus of Swahili is the Helsinki Corpus of Swahili [16]. This corpus has 12 million words from various texts. Despite few public domain NLP tools for Swahili, we still get some tools on open domains such as the Aflat site [17] and the TreeTagger toolkit [18,19]. More deliberate research effort is needed to increase the resources available for Swahili, including NLP toolkits such as those for QA, which are not available at all. This is a research gap that needs to be addressed. A candidate NLP solution for the QA task is the use of Semantic Networks.

Semantic networks model a problem domain as a network of nodes and edges, which can then be applied in different areas of NLP for tasks such as disambiguation, summarization, information extraction and question answering [20]. Using semantic networks is a viable option for developing QA systems for low resource languages which have few public domain corpora and NLP processing tools. It is possible to develop such networks by basic NLP processing e.g. parts of speech tagging (POST) then leverage on the language structure of subject-verb-object as the basic form of Swahili [15,21] to populate

the network. The network can then be used for QA tasks. The network can be subjected to queries using any typical database query language, such as SPARQL.

2. Objectives

This research aims at developing a semantic network from raw Swahili text corpora, then test the generated network on the task of Question Answering (QA. Swahili being a low-resourced language implies that the use of machine learning systems that require training data is not possible, due to lack of corpus to train on. The system for QA therefore must be developed from syntactic and semantic analysis of the raw text.

The significance of this research is the modelling of an information system from raw unstructured text to create a more useful semantic network. This network can then be useful for any query-based NLP task, including web-based search from an underlying Swahili corpus or even interactive systems that rely on underlying NL corpus e.g. chatbots and dialog systems. Typically, any querying task that relies on unstructured text corpus can benefit from this model. Alternative processing such as keyword search, n-gram search or tf-idf may not be suitable or possible for low-resource language, such as Swahili, considering the corpus sizes available for the QA task for Swahili language. The research also contributes to resourcing the Swahili language for benefit of the language itself and also for further researchers.

3. Methodology

The methodology adopted for this research borrows from work already done by the author on automatic semantic network generation from unstructured text [20]. The basic idea of developing the semantic network is to determine the resource description framework (RDF) triples from the text corpus. Generating these subject-predicate-object triples is done by an initial POST processing, followed by syntactic processing to determine how to extract subjects, predicates and objects from any POSTed text. Existing utilities such as Stanford CoreNLP [22] already have such ready for use pipelines in their open domain frameworks. However, these tools are only able to process English language text, and few other languages such as Arabic, Chinese, French, German, Spanish [22].

Getting subject-predicate-object triples out of the POSTed text must therefore be done using rules that are formulated to deal with the annotated text. Leveraging on the fact that Swahili follows the subject-verb-object (S-V-O) language structure, we can parse the POSTed text to look for, identify and retrieve these triples. These are the triples that we need for the development of the semantic network.

Once the triples have been generated, then it is possible to create the semantic network, which is usually represented in RDF syntax. Programming languages such as python also have RDF processing and visualization tools.

The question part of the QA system is processed in a similar manner, by first getting the POST, then identifying the query words. The query intention is to generate an enquiry about the subject or the object i.e. graph search on the nodes of the network. The query is presented to the semantic network as a SPARQL query.

The main challenge in testing the SN is the difficulty in obtaining any curated public domain QA datasets, unlike the English language domain where we have many QA sets such as BAbI [1], SQuAD [3], MCtest [2] etc. This research therefore uses public websites that have Swahili Frequently Asked Questions (FAQ), in Swahili known as ‘Maswali Yanayoulizwa Mara kwa Mara’ (MYM), as the basis of developing the semantic networks and gauging the possibility of responding to the question in the MYM. The particular MYM used for this research is the BBC MYM page [23].

The research also develops a semantic network from the Swahili story books obtained from the Tusome early literacy programme used by Kenyan grade one and two schools as released by the Kenyan Ministry of Education [24]. This was a project initiated in the 2011-2014 period and confirmed suitable for the free primary education scheme in the 2014-2019 period [25]. The whole Tusome project has 28 texts from series 41 to 69, though some of the texts have multiple stories within the same series.

The available texts in this corpus are suitable for early childhood education. Each text is accompanied a set of comprehension questions, mostly based on one fact from the story. It however has some element of inference since the responses may not be directly obtained from single sentences or n-grams in the story. This corpus can be thought to be of a similar setting as the MCtest set [2], though the range of stories in Tusome corpus is not as vast.

Experimental setup workflow:

1. Obtain text from corpus (for Tusome [24,25] the text is in PDF, or text areas on web pages for MYMs for BBC corpus [23])
2. Copy the text onto a text file e.g. corpus1raw.txt
3. Tag the raw text by setting up TreeTagger [18,19] to read the input text file e.g. corpus1raw.txt and then generate a tagged output file e.g. corpus1tagged.txt
4. For visualization, copy and save the output corpus as CSV e.g. corpus1tagged.csv. This file consists of a 3-column table showing the original tokens, the POST and the lemma.
5. Analyze the CSV file to formulate rules to generate subject-verb-object (S-V-O) triples on a top down basis for the whole text. Typical rules found to generate suitable candidates include N-VFIN-N triples, N-GEN-CON-N triples and substitution of N with PROPNAME in such triples
6. Create a python program script that reads the lemmas for the parts of speech identified in the rule set from step 5 above and harvest the qualifying list of S-V-O triples
7. The generated S-V-O triples are further enriched to create an RDF ready file to create a final output file e.g. in turle (TTL) format by adding the requisite prefix headers and tagging each lemma triple e.g. corpus1svo.ttl
8. The Question text is similarly processed to obtain S-V-O or S-V-?
9. SPARQL query is created based on the question triples in 8. to query the RDF corpus

4. Results

The research creates different semantic networks (the network where the answers shall be got), such as shown in Fig. 1 below, which is derived from the corpus of the book '42_C1_Somo_nilipendalo', for grade 1. We then query the SN (the question part) using SPARQL.

This corpus had five comprehension questions. A typical question was:
"*Q1 - Watoto wanapangwa katika vikundi vingapi? Taja majina ya vikundi hivyo*?"
(How many groups were the children arranged in? Name these groups).

This is answered by a SPARQL query that traverses the graph of Fig. 1 from the starred part (mtoto) and follows the black arrow towards the answer 'tatu' (three). For the second part of the question (names), the query traverses from 'kikundi' (group), then the 'jina' (name) node gives the names as 'nyuki', 'nyingu' and 'siafu'* (*not shown on Fig. 1 extract). A typical SPARQL query will be like the one below, having been obtained by POST processing of the question to get the S-V-O or S-V-? or ?-V-O of the question.

```
select ?o
where {
  :mtoto ?p :darasa .
  :darasa :pangwa :kikundi .
  :kikundi :ni ?o .
}
```

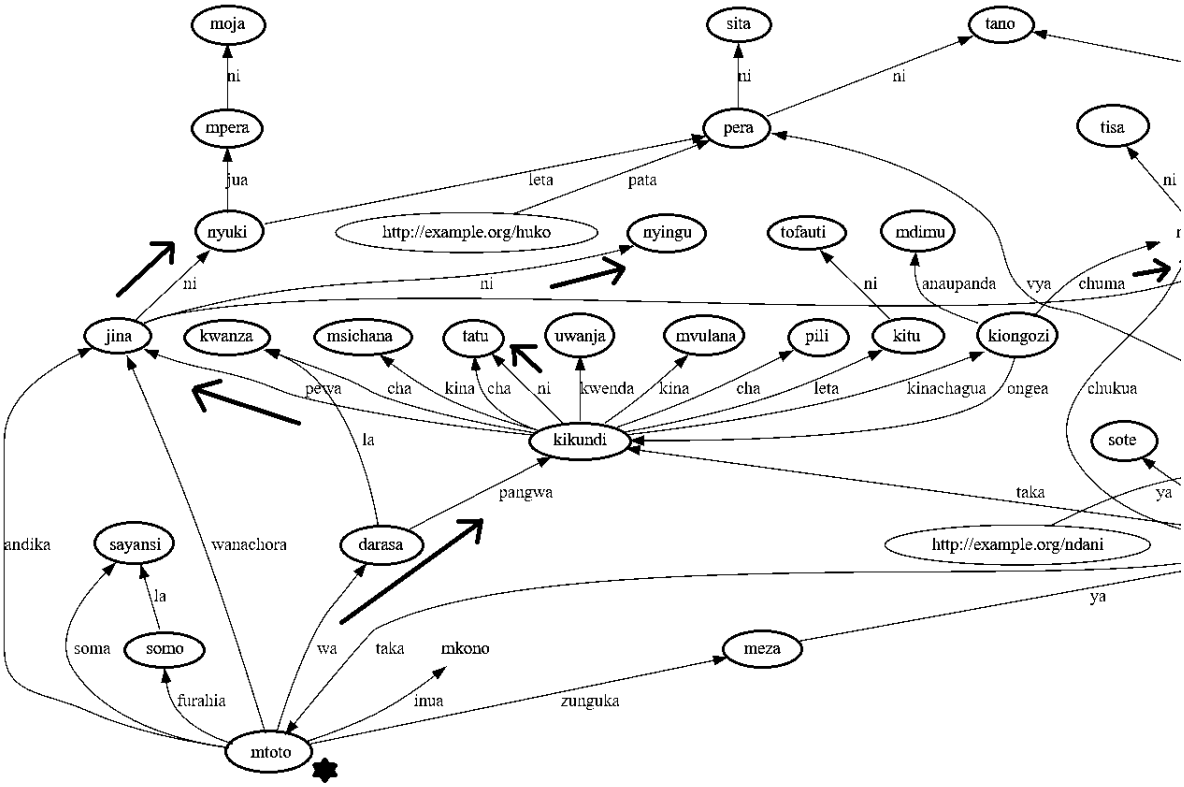

Fig. 1: Sample Semantic Network showing a query path (source: Author)

While the network could infer responses to four questions, it was not able to respond to the fifth question, "*Q5. Unafikiri watoto waliona nini ndani ya vipande walivyopewa na Mwalimu Nekesa*?" (In your opinion, what did the children see in the fruit pieces?). This is because of complexity of the question that asks for an opinion, whose answer cannot be directly inferred, though a human subject can reason through the story. The SN therefore achieves 4/5 correct answers for the story questions (80% accuracy) in this corpus.

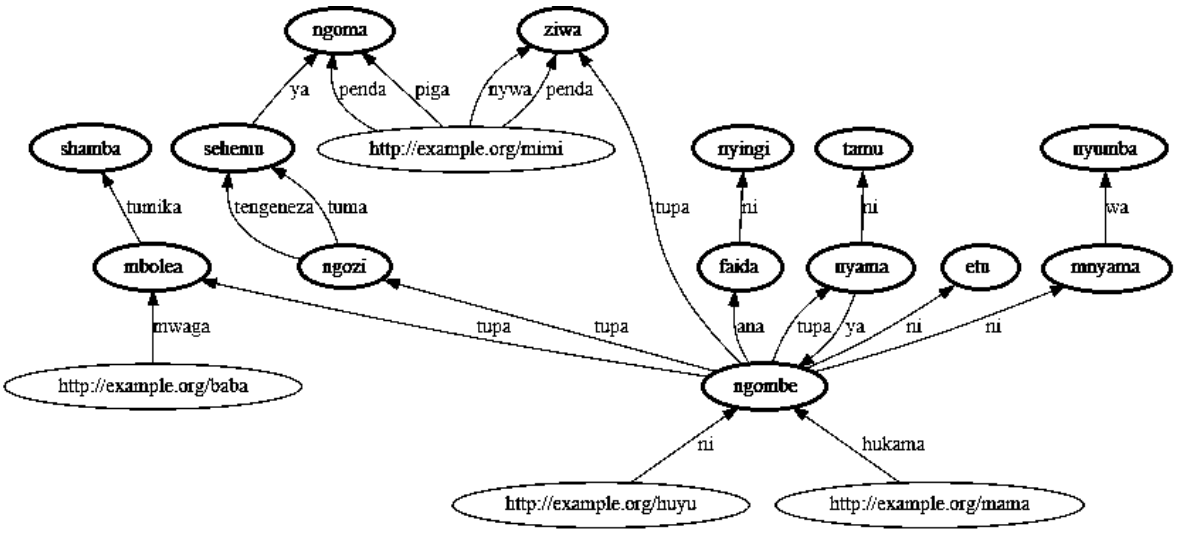

Fig. 2: Semantic Network from one full story of Tusome Corpus (source: Author)

Fig. 2 shows another SN for a different corpus being 53_C1_Faida_za_ngombe. This story had three questions, all answerable, 100% from the network after resolving that keyword 'pata' (get) as a question would mean the same as 'tupa' (gives) in the narrative.

The three questions were: '*Q1. Taja vitu vinne tunavyopata kutokana na ng'ombe*'. (State four benefits got from cows) '*Q2. Ngozi ya ng'ombe hutumiwa kutengeneza nini*?'

(What is hides used for?) '*Q3. Mbolea ya ngombe hutumika wapi*?' (Where is manure used?). On average, the different SN generated for the different corpora could answer at least 3 of every 5 questions in the 25% sample texts tested.

For the MYM corpus, the raw text [23] was processed by TreeTagger [18,19] and the candidate S-V-O triples were identified and extracted by our rules. These triples were then used as the final RDF triples for creating the semantic network. As an example, one MYM question was, "GMT ni nini?" (what is GMT?). This can be answered by the semantic network. However, the GMT bit had many proper nouns to track, due to lack of named entity recognition (NER) mechanisms, which would really have helped the processing.

5. Discussions

The results of the experiments show that it is possible to generate S-V-O triples from a typical natural language corpus of Swahili. These S-V-O triples can then be used to model the language structure, giving it meaning by virtue of the interconnections between nodes and the linkages of this knowledgebase. Additionally, the objective of a typical QA task is to query the linkages within the graph for relationships between concepts (subjects and objects) directly or indirectly. An advantage of using semantic networks is that the nodes can be subjects or objects, hence a query task is just an enquiry into the nodes of the network. It is the interconnectedness that provides the advantage that is exploited in the QA task. Alternative QA approaches e.g. tf-idf, n-gram search or keyword search seem unsuitable for low-resource corpus that typically have short text corpora with little relationships between keywords.

The results indicate that the semantic network can model the Swahili language corpus by creating a network of subjects, objects and their interrelatedness. This knowledgebase is suitable and ready for QA tasks, where answers are enquiries into the relatedness of S-P-O of the network. Challenges noted in creating the network include inability to pick out all the candidate S-V-O triples. This problem is mainly due to inadequacies of the pre-processing tools and the limited range of rules to pick out the final S-P-O triples.

For the two corpora, the tool used (TreeTagger) fails to recognize proper nouns and named entities in the Tusome corpus, giving it the <unknown> tag. This can be resolved by analysis of how the tagger performs on various corpora and then formulating a rule for named entity recognition (NER) in the absence of an alternative toolset for this. The BBC corpus [23] is noted to be a bit complex in language structure, unlike the Tusome corpus [24,25]. There may be need for some additional pre-processing tasks before generating the final candidate S-V-O e.g. NERs, effect of brackets, effects of commas and the need for simplification of sentences etc.

Formulating rules to pull the final S-V-O triples for creating the semantic network can become complex and lack in the expected generalization that should be applicable to most corpus. In established NLP tools such as Stanford CoreNLP [22], many of the rules are incorporated in the toolkit, making the tool to be generic enough for most cases.

Aspects such as named entity recognition (NER) and coreference resolutions is currently difficult in the absence of an existing toolkit for Swahili corpus. Stanford CoreNLP [22] has such tools but for English and other languages, but not Swahili. Long sentences with conjunctions ('pia'/also) and commas are also difficult to track without coreferencing. Another challenge was that TreeTagger did not process negation correctly i.e. 'hawakubaki' (did not remain) results into the lemma 'baki' (remain) without any negation tag. Such a glitch can be difficult to detect and mitigate. There is a need to do a full analysis of such shortcomings and reprocess these in some other ways e.g. new rules. Rule-based processing also suffer the disadvantage of increased number of rules that may be formulated to suite input texts. However, sticking the S-V-O structure model can simplify the rulesets.

Another difficulty with the QA system is synonym resolution. The question words may not necessarily be the words in the text. Some way of knowing the comparable corpus word that is equivalent to the Question word needs to be formulated e.g. on corpus 53_C1_Faida_za_ngombe, the question on '*Q1. Taja vitu vinne tunavyopata kutokana na ng'ombe*.' (Name four benefits from cows). The corpus does not have the key word 'pata' (get) on the text. The corpus instead has 'tupa' (gives). There is need to resolve that these two concepts are similar, in order to successfully query the SNs using SPARQL.

The current rule set that we are developing has not yet been subjected to a bigger corpus to test and confirm coverage. However, provided there is a way of generating S-V-O from the Swahili corpus, then the semantic network method for QA becomes realizable. While QA using SNs has been tried in other resource rich languages, few studies have been done on low resource languages, hence this research tries to confirm if SNs can be applicable successfully to Swahili, which is the novel finding, since this is confirmed empirically as being possible.

6. Conclusions

This research provided the preliminary results of using semantic networks as a basis for Question-Answering tasks, specifically for the under-resourced languages of Swahili. Swahili was chosen due to its major place as a language of communication in East and Central Africa, despite it having comparatively low public domain corpora for training, testing or even gold standard sets. Obtaining NLP tools for full pipeline processing was also difficult. Missing utilities included coref resolution and NER. However, using the knowledge of the Swahili language structure as subject-verb-object, the processed POST text was mined for relevant S-V-O triples, using formulated rules. These extracted triples were then used to generate the RDF triples needed for the semantic network. This network is then ready for NLP tasks.

The semantic network is found capable of the NLP task of QA. Experiments done on the QA tasks such as the FAQ corpus of the Swahili BBC website and Swahili storybook articles for lower primary schools in Kenya (Tusome corpus), show that the SNs can provide responses to typical questions. Difficulties in obtaining correct answers are mainly due to pre-processing challenges such as lack of coreference resolution or missing some rules that can be appropriate in the language context to extract the S-V-O triple. Natural language is also dynamic, and the structure may not necessarily conform to expected rules of formal grammar. Parsing such language may not always result into intelligible formal language presentation needed in SNs. More work needs to be done in terms of further experimentation on larger corpus and texts of varied complexity. The preliminary results are promising and can position Swahili as a more resourced language for the benefit of the speakers and researchers. By extension, this method should be applicable to the NLP task of QA for any S-V-O structured language.

References

[1] J. Weston, A. Bordes, S. Chopra, A.M. Rush, B. van Merriënboer, A. Joulin and T. Mikolov, Towards AI-complete question answering: A set of prerequisite toy tasks, arXiv preprint arXiv:1502.05698, 2015.

[2] M. Richardson, C.J. Burges and E. Renshaw, MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text, pp. 193-203, EMNLP, 2013.

[3] P. Rajpurkar, J. Zhang, K. Lopyrev and P. Liang, Squad: 100,000+ questions for machine comprehension of text, arXiv preprint arXiv:1606.05250, 2016.

[4] S. Wang and J. Jiang, Machine comprehension using match-LSTM and answer pointer, arXiv preprint arXiv:1608.07905, 2016.

[5] P. Gupta and V. Gupta, A survey of text question answering techniques. In: International Journal of Computer Applications, Vol. 53, No. 4, 2012.

[6] A. Mishra and S.K. Jain, A survey on question answering systems with classification, Journal of King Saud University-Computer and Information Sciences, vol. 28 no. 3, pp. 345-361, 2016
[7] E. Hovy, L. Gerber, U. Hermjacob, M. Junk and C. Lin, Question answering in webclopedia. In: Ninth Text Retrieval Conference, Volume 500–249 of NIST Special Publication, Gaithersburg, MD, National Institute of Standards and Technology, pp. 655–664, 2000.
[8] O.S. Goh and A. Cemal, Response quality evaluation in heterogeneous question answering system: A black-box approach, World Academy of Sciences Engineering and Technology, 2005.
[9] S.K. Vanitha and I.M. Lakshmi, Approaches for question answering system, IJEST, vol. 3, pp. 992–995, 2010.
[10] L. Yao, C. Mao and Y. Luo, Graph convolutional networks for text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 7370-7377, 2019.
[11] N. De Cao, W. Aziz and I. Titov, Question answering by reasoning across documents with graph convolutional networks, arXiv preprint arXiv:1808.09920, 2018.
[12] V. Berment, Me´thodes pour informatiser des langues et des groupesde langues peu dote´es, Ph.D. Thesis, J. Fourier University – GrenobleI, 2004.
[13] L. Besacier, V.B. Le, C. Boitet and V. Berment, ASR and translation for under-resourced languages. In: 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, vol. 5, pp. V-V, IEEE, 2006.
[14] J. Mendelson, P. Oplustil, O. Watts and S. King, Nativization of foreign names in TTS for automatic reading of world news in Swahili, 2017.
[15] H. Gelas, L. Besacier and F. Pellegrino, Developments of Swahili resources for an automatic speech recognition system. In: Spoken Language Technologies for Under-Resourced Languages, 2012.
[16] A. Hurskainen, Helsinki corpus of swahili. Compilers: Institute for Asian and African Studies (University of Helsinki) and CSC, 2004.
[17] Aflat, "Kiswahili Part-of-Speech Tagger – Demo", https://www.aflat.org/swatag, Retrieved 02 Dec 2019.
[18] H. Schmid, Improvements in Part-of-Speech Tagging with an Application to German. In: Proceedings of the ACL SIGDAT-Workshop. Dublin, Ireland, 1995.
[19] TreeTagger, "TreeTagger - a part-of-speech tagger for many languages", https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/, Retrieved 02 Dec 2019.
[20] B.W. Wanjawa and L. Muchemi, Automatic Semantic Network Generation from Unstructured Documents - The Options. In: 2018 5th International Conference on Soft Computing & Machine Intelligence (ISCMI), pp. 72-78, IEEE, 2018
[21] N.M. Ndung'u, Information Structure in Kiswahili, In: International Journal of Education and Research, vol. 3 no. 3, 2015.
[22] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard and D. McClosky, The Stanford CoreNLP Natural Language Processing Toolkit. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60, 2014.
[23] BBC News Swahili, "Usaidizi na Maswali yaulizwayo mara kwa mara", https://www.bbc.com/swahili/taasisi/2010/04/000000_help, Retrieved 02 Dec 2019.
[24] Kenya Ministry of Education, "Tusome", http://www.education.go.ke/images/Project-KPED/Brief%20on%20TUSOME%20.pdf, Retrieved 02 Dec 2019.
[25] B. Piper, J. Destefano, E. M. Kinyanjui and S. Ong'ele, Scaling up successfully: Lessons from Kenya's Tusome national literacy program. In: Journal of Educational Change, vol. 19 no. 3, pp 293-321, 2018.