Design and Development of an automated digital assessment system for virtual examination

Lalit Garg, Neha Agarwal, Abhinav Verma

Department of Computer Science and Engineering
Jaypee Institute Of Information Technology, Noida

Introduction

The project's title is "Design and development of digital assessment systems for virtual examination" in which we are trying to produce a solution for the modern problem, as the world is moving towards the automation and digital age, so there is a need for automation in the Indian examination system as well. The current pandemic has raised many questions, has shown how badly Indian educational institutions' are lacking in creating a fair environment for students and teachers as well. Major problem is to get a fair platform to automatically generate questions from a given syllabus, and checking of answers requires a high concentration, time, and energy which has increased the workload of teachers immensely and is prone to mistakes as well. Hence there is a need for an automatic system that can generate questions, check the answers, and generate performance. Also, managing a huge database of question-answer systems and different unfair means like paper-leakage, and generating different sets of questions is a major concern. In the manual system, it may be possible that different marks are given for the same answer. Solving all these problems, this system can lead to more efficient, fair, and, smooth conduction of examination.

The report describes the research for information on the recent work and solutions proposed to overcome the challenges in the automatic generation of question-answer for the development of an efficient digital assessment system for virtual examination. A question-answer is an efficient way of information retrieval. The objective of the research is to reduce time consumption and manpower in manually generating objective and subjective questions and automatically generating the responses by matching the human answer with the correct answer. Many researchers have proposed strategies for an automatic question-answer generation. After analyzing all the findings and optimization techniques in the research papers, we are able to conclude a final optimized examination assessment system. By this system, evaluation error in the marks will be reduced. We have started from the research on basic NLP pipeline tools using tokenization, lemmatization, stemming, structure tree parser, part of speech tagging and then found significant work on the improvement of part of speech tagging using optimization techniques like hidden Markov model (Viterbi Algorithm), Rule-based POS tagging, transformation-based tagging, removing lexical and syntactical ambiguities in the texts while retrieving information using text summarization techniques like TF-IDF algorithm for paraphrasing, finding an optimized set of answers synonyms, UML diagram to remove an attachment and homonymy syntactic ambiguity. We have also focused on the optimization of evaluating subjective answers using various methodologies.

Lierature Review

1. Algorithm for generating questions from the text

[6] The main goal of the framework is to evaluate the performance of the system using natural language processing tools and talk about some challenges faced. There are two phases- the first is the data preprocessing phase and the second is the generation of question-answer from the final data. Steps in the given process include- load the data of different subjects, detect potential sentences to be framed as the question where an important keyword like location, date, person, etc. can be framed as an answer, Eliminate unwanted phrases and punctuation marks, frame a grammar rule according to which question sound be framed, Parse the tree using Regular expression and store the desired question and answers. For Fill in the blank type of questions, the key point was to be able to identify the correct gap for which the blank needs to be created. The question generation system is built with an objective to semi-automate /automate the process to generate questions for exams and quizzes. Most of the system will be based on information retrieval and classic NLP algorithm tokenization, stemming, lemmatization, grammar, tree parser using grammar. It also talked about a novel application where a student can self-analyze their preparation using this model. Many online texts do not come with question-answer to analyze the reading and understanding skills of the student.

There were different lexical and syntactical challenges encountered in the paper. Attachment and Homonymy syntactic ambiguity where words in the sentence can have different meaning and thus Pos tagging become ambiguous and thus can give an error in the end questions framed. Paraphrasing is another challenge in generating questions where slight variation questions are expected from the text. Slightly different questions are important as it increases the difficulty level of the exam paper and helps in evaluating at a better level. It also talked about problems in evaluating subjective answers where some model is required to maintain connections between the sentences in the text.

2. Generating Multiple Choice test from a Medical Text: A Pilot Study

[8] The research talked about a pilot study on the generation of multiple-choice test questions from a text for medical text. Although a multiple-choice test is a common evaluatory process, it is a tedious task for teachers. Mitkov in 2006 developed a system that retrieves essential information from the text. This gave rise to a relatively new research area within the emerging field of Text-to-Text Generation (TTG) called Multiple-Choice Test Item Generation (MCTIG). Analysis of MCTIG produced semi-automatic generation and its use in the classroom reveals that educational value is not compromised. In fact, time and labor are saved.

Based on Mitkov, the process is divided into four phases- Parsing, Key-term identification, Source clause selection, Transformation to stem.

Sentence Parsing-Sentence Parsing is an important area as it checks the sentence fit for the given grammar.RIG employs Charniak's (1997) parser which appeared to be quite robust in the medical domain.

Key-Term Identification- MCTI should have a key-term as its answer rather than irrelevant concepts. For example, in the sentence To which disease or syndrome may chronic hepatitis progress if it is left untreated? The word 'chronic hepatitis' are quite prominent and can be considered as a key term which in turn means the sentences having key term should be filtered out to frame the question. The tf.idf method is used to promote the 30 most prominent potential key terms within the source text for subsequent processing, ruling out generic terms such as "patient" or "therapy" which are very frequent within a larger collection of medical texts.

Source Clause selection-

Mitkov filtered the clause in the text focusing on the rules and listed inappropriate structures for MCTIG (key terms underlined)

1. Subordinate clause-Although chest pain is a problem.

2. Negated clause- Ram should not play outside.
3. Coordinated NP- Excessive tension causes mental illness and blood pressure problems.
4. Initial Pronoun- It associates with hypertension instead.

After taking care of above listed inappropriate structures and filtering the appropriate source clause, a finite main clause which contains an NP headed by a key term and functioning as a source or object with all the subordinate clauses provided that it does not contain the inappropriate structures listed are found eligible for the source clause.

Transformation to stem-

Experimentation during development showed that our model improves by 30 compared to the baseline approach. After the identification of the source clause, it has to b turned to the stem of an MCTI. A stemming algorithm reduces the words “chocolates”, “chocolatey”, “choco” to the root word, “chocolate” and “retrieval”, “retrieved”, “retrieves” reduce to the stem “retrieve”. Stemming is an important part of the pipelining process in Natural language processing.

Selection of Appropriate Distractors-

MCTIs aim to select appropriate distractors which are important in generating a better level evaluation assessment. An appropriate distractor is a concept semantically close to the answer however cannot serve as a right answer itself.

Three experts in producing MCTIs for medical assessment jointly reviewed 279 MCTIs (each featuring four distractors) generated by the system. Three chapters from a medical textbook served as the source texts while a much larger collection of MEDLINE texts was used as the reference corpus.

3. SINGLE DOCUMENT AUTOMATIC TEXT SUMMARIZATION USING TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY (TF-IDF)

[5] The increasing availability of online information has triggered intensive research in the area of automatic text summarization within the Natural Language Processing (NLP). Text summarization reduces the text by removing the less useful information which helps the reader to find the required information quickly. There are many kinds of algorithms that can be used to summarize the text. One of them is TF-IDF (Term Frequency-Inverse Document Frequency). This research aimed to produce an automatic text summarizer implemented with the TF-IDF algorithm and to compare it with other various online sources of automatic text summarizer.

The increasing availability of online information has triggered intensive research in the area of automatic text summarization within the Natural Language Processing (NLP). Text summarization reduces the text by removing the less useful information which helps the reader to find the required information quickly. There are many kinds of algorithms that can be used to summarize the text. One of them is TF-IDF (Term Frequency-Inverse Document Frequency). This research aimed to produce an automatic text summarizer implemented with the TF-IDF algorithm and to compare it with other various online sources of automatic text Summarizer. These steps have their respective tasks. First, preprocessing consists of the operation needed to enhance feature extraction such as tokenization, part of speech tagging, removing stop words, and word stemming. Second, it is feature extraction. It is used to extract the features of the document by obtaining the sentence in a text document based on its importance and given the value between zero and one. Third, sentence selection and assembly are when the sentences are stored in descending order of the rank, and the highest rank is considered as the summary. Last, summary generation is the sentences that are put into the summary in the order of the position in the original document.

TF=total count of words in a document/total number of words in a document IDF=log (All document Number/Document frequency)

Finally, three to five sentences with the highest TF-IDF value are chosen. The number of sentences in the final summary may change depending on the compression rate of the program chosen by the user. As TF-IDF is an extraction method, the sentences that appear in the summary are the same as the original document. These chosen final sentences are sorted in accordance with their appearance in the original document.

4. A Study on Different Part of Speech(POS) Tagging Approaches in Assamese Language

[13] Syntactic parsing is a necessary task that is essential for Natural language processing methodology including Part of Speech (POS) tagger. For the development and enrichment of languages, part of speech tagging plays a very crucial role. Part of speech tagging, especially for the regional Indian languages can give an international and worldwide approach. For a regional language like Assamese which is Assam's official language, part of speech tagging has become very much essential for the overall flourishment of the language. The technique of assigning an appropriate part of speech tag for each word in an input sentence of a language is called Part of Speech Tagging. It is commonly referred to as POS tagging.

The architecture of POS Tagger

1. Tokenization

The given text is divided into tokens so that they can be used for further analysis. The tokens may be words, punctuation marks, and utterance boundaries.

2. Ambiguity look up and Resolution

If an expression (word/phrase/sentence) has more than one interpretation, we can refer to it as ambiguous. The process to remove the ambiguity of words in a given context is called disambiguation. Disambiguation is based on information about words such as the probability of the word. For for example, the word "power" is more likely used as a noun than as a verb. Disambiguation is also based on contextual information or word/tag sequences. For example, the model might prefer noun analyses over verb analyses if the preceding word is a preposition or article. Disambiguation is the most difficult problem in tagging.

POS TAGGING TECHNIQUES :

1. Rule-based Tagging

Rule-based part-of-speech tagging is the oldest approach that uses hand-written rules for tagging. Rule-based tagger depends on a dictionary or lexicon to get possible tags for each word to be tagged. Hand-written rules are used to identify the correct tag when a word has more than one possible tag. Disambiguation is done by analyzing the linguistic features of the word, its preceding word, its following word, and other aspects. For example, if the preceding word is article then the word in question must be a noun. This information is coded in the form of rules.

2. Markov Model

The Markov model is nothing but a finite-state machine. Each state has two probability distributions: the probability of emitting a symbol and the probability of moving to a particular state. From one state, the Markov model emits a symbol and then moves to another state.The objective of Markov model is to find optimal sequence of tags T = t1, t2, t3,...tn for the word sequence W = w1,w2,w3,...wn. That is to find the most probable tag sequence for a word sequence.

3. Viterbi Algorithm/ Hidden Markov Models (HMM) in POS tagging

The Viterbi algorithm is a dynamic programming algorithm for finding the most likely sequence of hidden states – called the Viterbi path – that results in a sequence of observed events, especially in the context of Markov information sources and hidden Markov models. In an HMM, we know only the probabilistic function of the state sequence. At the beginning of the tagging process, some initial tag probabilities are assigned to the HMM **4. Transformation-based learning**

Transformation-based learning (TBL) is a rule-based algorithm for automatic tagging of parts-of-speech to the given text. TBL transforms one state to another using transformation rules in order to find a suitable tag for each word. TBL allows us to have linguistic knowledge in a readable form. It extracts linguistic information automatically from corpora.

Parsing

Parsing is another important aspect utilized in conjunction with part-of-speech tagging to identify and understand natural language sentences. With parsing, when given an input sentence and a grammar, it can be determined whether the grammar can generate the sentence. Parsing can be described, at least in this context, as “the process of analyzing a string of words to uncover its phrase structure, according to the rules of the grammar”.

5. Hidden Markov Models

[3] The Hidden Markov Model (HMM) is a popular statistical tool for modeling a wide range of time-series data. In the context of natural language processing(NLP), HMMs have been applied with great success to problems such as part-of-speech tagging and noun-phrase Chunking. The Hidden Markov Model(HMM) is a powerful statistical tool for modeling generative sequences that can be characterized by an underlying process generating an observable sequence.
HMMs have found application in many areas interested in signal processing, and in particular speech processing, but have also been applied with success to low-level NLP tasks such as part-of-speech tagging, phrase chunking, and extracting target information from documents. Andrei Markov gave his name to the mathematical theory of Markov processes in the early twentieth-century, but it was Baum and his colleagues that developed the theory of HMMs in the 1960s.

A is a transition array, storing the probability of state j following state i . Note the state transition probabilities are independent of time:
A = [a ij] , a ij = P (q t = s j —q t1 = s i) .
B is the observation array, storing the probability of observation k being produced from the state j, independent of t:
B = [b i (k)] , b i (k) = P (x t = v k —q t = s i) .
is the initial probability array:
= [i] , i = P (q 1 = s i) .
Two assumptions are made by the model. The first, called the Markov assumption, states that the current state is dependent only on the previous state, this represents the memory of the model. Given an HMM, and a sequence of observations, we’d like to be able to compute P (O—), the probability of the observation sequence given a model. This problem could be viewed as one of evaluating how well a model predicts a given observation sequence, and thus allow us to choose the most appropriate model from a set.

6. Adaptation, Comparison, and Improvement of Metaheuristic Algorithms to the Part-of-Speech Tagging Problem.

[15] Part-of-Speech Tagging (POST) is a complex task in the preprocessing of Natural Language Processing applications. Tagging has been tackled from statistical information and rule-based approaches, making use of a range of methods. Most recently, metaheuristic algorithms have gained attention while being used in a wide variety of knowledge areas, with good results. As a result, they were deployed in this research in a POST problem to assign the best sequence of tags (roles) for the words of a sentence based on information statistics. This process was carried out in two cycles, each of them comprised four phases, allowing the adaptation to the tagging problem in metaheuristic algorithms such as Particle Swarm Optimization, Jaya, Random-Restart Hill Climbing, and a memetic algorithm based on Global-Best Harmony Search as a global optimizer, and on Hill Climbing as a local optimizer. In the consolidation of each algorithm, preliminary experiments were carried out (using cross-validation) to adjust the parameters of each algorithm and, thus, evaluate them on the datasets of the complete tagged corpus: IULA (Spanish), Brown (English) and NasaYUWE (Nasa). The results obtained by the proposed taggers were compared, and the Friedman and Wilcoxon statistical tests were applied, confirming that the proposed memetic, GBHS Tagger, obtained better results in precision. The proposed taggers make an important contribution to POST for traditional languages (English and Spanish), non-traditional languages (Nasa Yuwe), and their application Areas.

Consequently, the presented research reinforced the idea that metaheuristic approaches are capable of performing tagging with good results, with acceptable resources and times. Metaheuristic algorithms should continue to be used for tagging on other traditional and non-traditional languages, and seek new improvements for the proposed taggers in combination with other optimization techniques that improve the results of the tagging.

7. Automatic Generation of Assessment Test Items from Text: Some Quality Aspects

[9] The general idea is to extract fragments from the source text document and to transform them into questions or test items. For instance, Heilman has discovered numerous challenges in question generation from the text. These include linguistic challenges (lexical, syntactic, discourse-related) as well as various challenges related to the application of question generation tools in classrooms.
1. Text preprocessing — to convert a raw text file into a well-defined sequence of linguistically meaningful units, or segments
2. Segment filtering — to filter the set of segments so that it contains the most salient segments
3. Test item generation — to transform the text segments into test items.
It is obvious that not every text sentence is appropriate for test item generation. We assume proper filtering of acquired sentences could have a convincing impact on the quality of the resulting test items set, and we propose using extractive text summarization to filter out the unnecessary text portions. In NLP, different methods for scoring sentences by importance are applied (usually in combination): sentence length cut-off (short sentences are excluded), use of cue phrases (inclusion of sentences with phrases such as “in conclusion”), sentence position in a document/paragraph, the occurrence of frequent terms (based on TF- IDF term weighting), and occurrence of title words.

Another issue, which arises at this step, is that the processed sentences may contain anaphora. Without the implementation of automatic anaphora resolution, the user could resolve the anaphora manually (e.g. to replace pronouns with corresponding nouns) using the in-context display of the processed sentence.

8. Generating Natural Language Questions to Support Learning ON-Line

[10] Natural language processing technology can be used to automatically generate such questions but techniques used have not fully leveraged semantic information contained in the learning materials or the full context in which the question generation task occurs. We introduce a sophisticated template-based approach that incorporates semantic role labels into a system that automatically generates natural language questions to support online learning. The methods we have labeled “semantics-based” use method(s) of target identification that are primarily semantic, using techniques such as semantic role labeling (SRL). Given a sentence, a semantic role labeler identifies the predicates (relations and actions) along with the semantic entities associated with each predicate. Semantic roles, as defined in PropBank include Arg0, Arg1, ..., Arg5, and ArgA. A set of modifiers is also defined and includes ArgM-LOC (location), ArgM-EXT (ex-tent), ArgM-DIS (discourse), ArgM-ADV (adverbial), ArgM-NEG (negation), ArgM-MOD (modal verb), ArgM-CAU (cause), ArgM-TMP (time),ArgM-PNC (purpose), ArgM-MNR (manner), and ArgM-DIR (direction).

In the content selection stage, a single sentence is first parsed with a semantic role labeler to identify potential targets. Targets are selected using simple selection criteria. Any of the predicate-specific semantic argument, if present, are considered. For example, AM-LOC can be used to generate a where question, and an AM-TMP can be used to generate a when question. After targets have been identified, these, along with the complete SRL parse of the sentence are passed to the question formulation stage. Two heuristics are used to rank the generated questions. Questions are ranked first by the depth of their predicate in the dependency parse of the original question. This is based on the assumption that questions arising from the main clauses are more desirable than those generated from deeper predicates. In the second stage, questions with the same rank are re-ranked according to the number of pronouns they contain, with questions with fewer pronouns having a higher rank.

The research has shown how a template-based method, using predominately semantic information, can be used to generate natural language questions for use in an online learning system. Our templates are based on semantic patterns, which cast a wide syntactic net and a narrow semantic net. The template mechanism supports rich selectional and generational capabilities, generating a large pool from which questions for learners can be selected.

9. Resolving Syntactic Ambiguities in Natural Language Specification of Constraints

[2] The research has identified a few cases where the Stanford parser is not able to handle particular syntactic ambiguities such as attachment ambiguity and homonymy. As a result of wrong syntax analysis, the semantic analysis goes wrong, and finally, the wrong question is generated. The identified cases of syntactic ambiguities not resolved by the Stanford parser are discussed and also a novel technique to automatically resolve the identified cases of syntactic ambiguities in English specification is presented. Following are the details of both types of syntactic ambiguity:

1. Attachment Ambiguity Attachment ambiguity is a type of syntactic ambiguity where a prepositional phrase or a relative clause in a sentence can be lawfully attached be prepwith(pay-2, bonus-9) to represent the actual meanings of the example i.e. the pay with bonus is given to all the employees.

2. Homonymy In linguistics, homonymy is a type of syntactic ambiguity in which a word in a phrase or a sentence exhibits different syntactic representations in different cases. For example, in the sentence 'A customer books two items', token 'books' is wrongly judged as 'NNS' by the Stanford parser. However, the token 'books' is a verb and the correct POS tag is 'VBZ'.ed to one of the two parts of that sentence.

Solution for Resolving Syntactic Ambiguity

For generating correct dependencies of input English sentences, we again use the information on hand in the input UML class model. As attachment ambiguity is due to the ambiguous role of noun with a preposition in a sentence. To correctly identify the attachment of the noun with the other two nouns, we map the (three) candidate English elements (such as nouns) to the classes in the UML class model.If the token matches to an operation-name or a relationship name then it is a verb or if the ambiguous token matches to a class-name or attribute-name then it is classified as a common noun or proper noun.

10. Answer Evaluation using Machine Learning

[1] Manual answer evaluation is a very tedious task. Manual checking is a very time-consuming process and also requires lots of manpower. Also, paper checker is not able to give marks equally. So, our system will evaluate answers based on some keyword, and also manpower will be saved. Only one has to scan the paper then, based on the keyword in the answer the system will provide the marks to the question according to the dataset present. Also, By this system, the evaluation error of the marks to the particular question will be reduced. So, our system will evaluate answers based on some keyword, and also manpower will be saved. Only one has to scan the paper then the system will split the answer using OCR, based on the keyword in the answer the system will provide the marks to the question according to the dataset present, There is a need for such an application which will provide an easy evaluation of answer and can provide eligible marks. The algorithm will evaluate the answer based on the length of the answer and important keywords covered which are specified by the teacher with each answer which is to be evaluated.
The algorithm assigns marks on basis of :
The Number of keywords matched and the Length of the answers.
The proposed system can evaluate 5760 answers in a day whereas a human working for 8 hours

can evaluate 480 answers a day. The scores are calculated for 10 students.The difference between manual evaluation and system evaluation is very close.

11. Grading Descriptive Answer Scripts using Deep Learning (Neethu George, Sijimol PJ, Surekha Mariam Varghese)

[14] One of the significant parts of education is the examination which is a measure of students learning ability. After examination, the teachers spend most of their time evaluating the marks of the students and the evaluation takes bulk usage of human effort, time, and cost. An automated assessment evaluation system can reduce the efforts during the evaluation.

The objective of this model is to extract the semantics to efficiently represent the text in answer scripts and develop a model from the key and evaluated answer scripts to grade non evaluated answer scripts using deep neural networks. It is a combination of NLP and machine learning. The learning is done by using the Recurrent Neural Network and LSTM cells. Deep Neural networks are able to capture the semantics of text in order to and the similarity between texts. The goal of the system is to replace the traditional human evaluation of the answer sheet that depends on several factors such as time, mindset, presentation style, and so on.

The embedding vector from the LSTM layer will be the semantic representation of the answer. Based on this value the output layer, fully connected neural network layer, the dense layer will predict the one- hotted score. Supervised training is used for this sequential model. The neural networks are called from Keras library. The final layer will then predict the score. The proposed Deep Descriptive Answer Scoring model (D-DAS model) is a sequential model that consists of an embedding layer, LSTM-RNN layer, dropout layer, and dense layer. The dense layer gives the one-hot encoded score for each answer. The proposed system is an automated descriptive answer checking and grading application using deep learning. For semantics interpretation and grading of descriptive answers, natural language processing, and deep learning tools have been used.

12. Automated Essay Scoring

[7] Revision and feedback are essential aspects of the writing process. Students need to receive feedback in order to increase their writing quality. However, responding to student papers can be a burden for teachers. Particularly if they have a large number of students and if they assign frequent writing assignments, providing individual feedback to student essays might be quite time-consuming. Four types of AES systems, which are widely used by testing companies, universities, and public schools: Project Essay Grader (PEG), Intelligent Essay Assessor (IEA), E-rater, and IntelliMetric.

Project Essay Grader (PEG)

Project Essay Grader (PEG) was developed by Ellis Page in 1966 upon the request of the College Board, which wanted to make the large-scale essay scoring process more practical and effective (Rudner Gagne). PEG uses proxy measures to predict the intrinsic quality of the essays. Proxies refer to the particular writing construct such as average word length, essay length, number of semicolons or commas, and so on (Kukich, 2000; Chung O'Neil, 1997; Rudner Gagne, 2001).

Intelligent Essay Assessor (IEA)

Another AES system, Intelligent Essay Assessor (IEA), analyzes and scores an essay using a semantic text analysis method called Latent Semantic Analysis (LSA) (Lemaire Dessus, 2001). Latent Semantic Analysis (LSA) is defined as “a statistical model of word usage that permits comparisons of the semantic similarity between pieces of textual information”.LSA first processes a corpus of machine-readable language and then represents the words that are included in a essay.

13. Automatic Assessment of Descriptive Answers for Online Examination using Semantic Analysis.

[12] Question Answering is a specialized form of information retrieval. Given a collection of documents, a Question-Answering system attempts to retrieve correct answers to questions posed in natural language. The proposed system includes components of data preprocessing, data normalization, stop word removal, porter stemmer algorithm, TF-IDF calculation, etc. These components play a major role in feature extraction and feature selection.

Finally, the system accuracy is determined with accuracy as well as a false-positive ratio. The system is composed of heterogeneous modules that work together to help the system attain its objective. Once an answer has been identified, the shallow parsing performed is leveraged to extract only the relevant words or phrases in answer to the question. The use of a part-of-speech tagger helps to enable recognition of answers of candidates with respect to the identified model answer. Answers of candidates can be ranked based on measures of distance between keywords, numbers of keywords matched, and other similar heuristic metrics.

This model will classify all answers based on similarity weights. Confidence in the correctness of an answer can be increased in a number of ways. One way is to use a lexical resource like WordNet (Synonyms) to verify that a candidate response is of the correct answer type. Using ANN evaluation of the marks is done automatically according to the current weights (range will be 0.01 to 0.99). The system uses NLP(Natural language processing) approach during the preprocessing phase and ANN(Artificial neural network) for the generation of similarity weights for the answers of candidates. Accordingly, the gained weight system will assign proportional marks to a specific answer.

14. Synonyms Paraphrasing using WordNet and Internet

[4] The paper proposes a method of synonymous paraphrasing of a text-based on WordNet synonymy data and Internet statistics of stable word combinations (collocations). Given a text, we look for words or expressions in it for which WordNet provides synonyms and substitutes them with such synonyms only if the latter form valid collocations with the surrounding words according to the statistics gathered from the Internet.
Synonymous paraphrasing (SP) is such a change of natural language (NL) text or of its fragments that preserves the meaning of the text as a whole. Nearly every plain text admits SP (in contrast to lists of names, numerical data, poetry, and the like). Computational linguistics has always considered SP an important and difficult problem. In this paper, a method of local SP of NL texts based on WordNet synonymy information (synsets) and Internet-based statistics on stable word combinations (collocations) is proposed. To paraphrase a text, we look for words or multi-words in it that are members of a WordNet synset and substitute them with other members of the same synset only if they are feasible components of collocations with the surrounding words according to statistical evaluation through the Internet search engine, such as Google.
Hereafter we assume that a set of synonymy tools is available that includes:
Synonymy dictionary such as WordNet (or EuroWordNet), A specially compiled dictionary of absolute synonyms that contain all the abovementioned types of English equivalents.
Various Types of Paraphrasing:
Text compression For this, the shortest synonym is taken in each synset (either independently of any statistical evaluations or selecting from the words that passed the marginality threshold). This gives a significant gain in space only when there are abbreviations (s) among absolute synonyms. **Text canonization** For this, the most frequently used synonym is taken. Of course, it may prove to be the same one as in the source text.t is also useful for persons with limited language knowledge, i.e. for foreigners or children, since this renders texts in a more intelligible way. **Text simplification** Any text will be more intelligible for a language-impaired person if we select among synonyms a “simpler”. It is not always the most frequently used synonym.

15. Extracting Word Synonyms from Text using Neural Approaches

[11] Extracting synonyms from textual corpora using computational techniques is an interesting research problem in the Natural Language Processing (NLP) domain. Neural techniques (such as Word2Vec) have been recently utilized to produce distributional word representations (also known as word embeddings) that capture semantic similarity/relatedness between words based on linear context. Nevertheless, using these techniques for synonyms extraction poses many challenges due to the fact that similarity between vector word representations does not indicate the only synonymy between words, but also other sense relations as well as word association or relatedness. In this paper, first, distributional word embeddings using Word2Vec build and then use the induced word embeddings as an input to train a feed-forward neutral network using an annotated dataset to distinguish between synonyms and other semantically related words. Word embeddings (also known as distributional word representations) are vector representations for words that are usually constructed from raw text based on linear context (words that occur in the neighborhood of a target word). In these representations, each word is converted into a vector of numerical values or real values. Computational techniques to extract synonyms from the raw text have been inspired by the classical distributional hypothesis “if two words have almost identical environments, we say that they are synonyms”.It addressed the synonymy identification problem as a classification task. To make the problem simpler and doable within the time-frame of the research, only adjectives were considered for the classification. The method used a feed-forward neural network with backward propagation as a learning algorithm. To obtain labeled training data, we extracted synonyms pairs from SimLex-999 similarity lexicon with a similarity score ¿ 6:5. SimLex-999 is a gold-standard resource for evaluating distributional semantic models

Summary

S.No	Name	Significance of work-Findings
1	Algorithm for generating questions from the Text	Natural text processing pipelines method Tokenization, Stemming, POS tagging, Structure tree Parser to generate a question-answer system from the text. Lexical and Syntactical Challenges were discussed like Paraphrasing, Pos tagging Ambiguity process, inefficiency to learn connections in the answer sentences.
2	Generating Multiple Choice tests from a Medical Text.	The method here gave us two key points to strengthen our NLP pipeline methods. Importance of answer in the fill in the blanks as a key term. Filtration of inappropriate structures for the main text chosen-Subordinate clause, Negated clause, Coordinated NP, initial pronouns.
3	Single Document Automatic Text Summarization Using Term Frequency-inverse Document Frequency (Tf-IDF)	Text summarization technique TF-IDF algorithm to overcome the challenge of paraphrasing(questions should have a slight variation from the original text) and dealing with complex text by first simplifying the text.
4	A Study on Different Part of Speech(POS) Tagging Approaches in Assamese Language	The technique of assigning an appropriate part of speech tag for each word in an input sentence of a language is called Part of Speech Tagging. It is commonly referred to as POS tagging. It is the main method in our naive process.

5	Hidden Markov Models	The detail on the optimization Model to remove ambiguity in POS tagging. Given an HMM, and a sequence of observations, we'd like to be able to compute P (O—), the probability of the observation sequence given a model
6	Adaptation, Comparison, and Improvement of Metaheuristic Algorithms to the Part-of-Speech Tagging Problem	Using metaheuristic algorithms like Particle Swarm Optimization, Hill Climbing in place of POS tagging for better results
7	Automatic Generation of Assessment Test Items from Text: Some Quality Aspects	Using text-preprocessing, segment filtering, test item generation, extract question fragments from the text were extracted
8	Generating Natural Language Questions to Support Learning ON-Line	Use of semantic-based methods to optimize framing question-segments. Given a sentence, a semantic role label identifies the predicates (relations and actions) along with the semantic entities associated with each predicate. A set of modifiers is also defined and includes ArgM-LOC (location), ArgM-EXT (ex-tent), ArgM-DIS (discourse), ArgM-ADV (adverbial), ArgM-NEG (negation), ArgM-MOD (modal verb), ArgM-CAU (cause), ArgM-TMP (time),ArgM-PNC (purpose), ArgM-MNR (manner), and ArgM-DIR (direction).For example, AM-LOC can be used to generate a where question, and an AM-TMP can be used to generate a when question.

9	Resolving Syntactic Ambiguities in Natural Language Specification of Constraints	Types of Syntactic Ambiguity faced:1. Attachment Ambiguity 2. Homonymy Ambiguity Resolution of Ambiguities: As attachment ambiguity is due to the ambiguous role of nouns with a preposition in a sentence. To correctly identify the attachment of the noun with the other two nouns, we map the (three) candidate English elements (such as nouns) to the classes in the UML class model. If the token matches to a relationship name then it is a verb or if the ambiguous token matches to a class-name or attribute-name then it is classified as a noun
10	Answer Evaluation using Machine Learning	The algorithm will evaluate the answer based on the length of the answer and important keywords covered which are specified by the teacher with each answer which is to be evaluated. The proposed system can evaluate 5760 answers in a day whereas a human working for 8 hours can evaluate 480 answers a day
11	Grading Descriptive Answer scripts using Deep Learning	It is a combination of NLP and machine learning. The learning is done by using the Recurrent Neural Network and LSTM cells. Deep Neural networks are able to capture the semantics of text in order to and the similarity between texts.

12	Automated Essay Scoring	Automated Essay Scoring Systems- Project Essay Grader uses average word length, essay length, number of semicolons, or commas to predict the intrinsic quality of essays. Intelligent Essay Assessor analyzes and scores an essay using a semantic text analysis method called Latent Semantic Analysis (LSA) that uses the semantic similarity between texts for evaluation
13	Automatic Assessment of Descriptive Answers for Online Examination using Semantic Analysis	Answers of candidates can be ranked based on measures of distance between keywords, numbers of keywords matched, and other similar heuristic metrics. This model will classify all answers based on similarity weights. Using ANN evaluation of the marks is done automatically according to the current weights (range will be 0.01 to 0.99).
14	Synonyms Paraphrasing using WordNet and Internet	Using a wordnet dictionary to produce synonyms of the word. Various types of paraphrasing-Text compression, Text canonization, Text simplification to fulfill the need to produce slightly deviated questions from the text. Another application is in producing synonyms of the correct answer so that different words with the same meaning as the correct answer should be evaluated correctly.

15	Extracting Word Synonyms from Text using Neural Approaches	Neural techniques (such as Word2Vec) have been recently utilized to produce distributional word representations (also known as word embeddings) that capture semantic similarity/relatedness between words based on linear context.

References

[1] Tariq Ahmad, Lars H Lund, Pooja Rao, Rohit Ghosh, Prashant Warier, Benjamin Vaccaro, Ulf Dahlström, Christopher M O'connor, G Michael Felker, and Nihar R Desai. Machine learning methods improve prognostication, identify clinically distinct phenotypes, and detect heterogeneity in response to therapy in a large cohort of heart failure patients. *Journal of the American Heart Association*, 7(8):e008081, 2018.

[2] Imran Sarwar Bajwa, Mark Lee, and Behzad Bordbar. Resolving syntactic ambiguities in natural language specification of constraints. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 178–187. Springer, 2012.

[3] Phil Blunsom. Hidden markov models. *Lecture notes, August*, 15(18-19):48, 2004.

[4] Igor A Bolshakov and Alexander Gelbukh. Synonymous paraphrasing using wordnet and internet. In *International Conference on Application of Natural Language to Information Systems*, pages 312–323. Springer, 2004.

[5] Hans Christian, Mikhael Pramodana Agus, and Derwin Suhartono. Single document automatic text summarization using term frequency-inverse document frequency (tf-idf). *ComTech: Computer, Mathematics and Engineering Applications*, 7(4):285–294, 2016.

[6] Shweta Devare Phadke, Sabina Kazi Hakim, Meera Deshmukh, and Pranati Tilak. Comparison of physiotherapy and management students' perception towards online examination introduced during covid-19 pandemic. 07:5403–5409, 08 2020.

[7] Semire Dikli. Automated essay scoring. *Online Submission*, 7(1):49–62, 2006.

[8] Nikiforos Karamanis, Ruslan Mitkov, et al. Generating multiple-choice test items from medical text: A pilot study. In *Proceedings of the fourth international natural language generation conference*, pages 111–113, 2006.

[9] Andrey Kurtasov. Automated generation of assessment test items from text: Some quality aspects. In *AIST (Supplement)*, pages 91–95, 2014.

[10] David Lindberg, Fred Popowich, John Nesbit, and Phil Winne. Generating natural language questions to support learning on-line. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 105–114, 2013.

[11] Nora Mohammed. Extracting word synonyms from text using neural approaches. *Int. Arab J. Inf. Technol.*, 17(1):45–51, 2020.

[12] Toshavi Patil. Automatic assessment of descriptive answers for online examination using semantic analysis. *Journal of the Gujarat Research Society*, 21(5):413–419, 2019.

[13] Bipul Roy and Bipul Syam Purkayastha. A study on different part of speech (pos) tagging approaches in assamese language. *International Journal of Advanced Research in Computer and Communication Engineering*, 5(3), 2016.

[14] PJ Sijimol and Surekha Mariam Varghese. Handwritten short answer evaluation system (hsaes). *IJSRST*, 4, 2018.

[15] Miguel Alexis Solano-Jiménez, Jose Julio Tobar-Cifuentes, Luz Marina Sierra-Martínez, and Carlos Alberto Cobos-Lozada. Adaptation, comparison, and improvement of metaheuristic algorithms to the part-of-speech tagging problem. *Revista Facultad de Ingeniería*, 29(54):e11762–e11762, 2020.