JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.6, Special Issue.10, Nov 2018 E-ISSN: 2347-2693

Using Proximity and Semantic Similarity in Question Answering

Shaunak S. Phaldessai[1*], Amey D. S. Kerkar [2]

[1]Student, Don Bosco College of Engineering, Goa, India
[2]Department of Computer Engineering, Don Bosco College of Engineering, Goa, India

[*]Corresponding Author: shaunak.phaldessai19@gmail.com, Tel.: +91-8007865727

Available online at: www.ijcseonline.org

Abstract— This paper deals with the process of Question Answering, using news articles crawled from 'THE HINDU' newspaper website of the year 2017. We make use of corpus of close to 10,000 articles/documents crawled categorically into Sports, Science and Tech., Business and Entertainment. We have implemented a system that extracts documents based on relevance to the question a user asks through the tf-idf ranking. For the processing phase, we made use of methods initially implemented for simpler systems, such as document extraction and checking sentence similarity between two short sentences. We managed to implement the techniques to extract coherent answers by extracting the passages with the best likelihood of containing the answer and the process these passages for the answer based on their similarity with the question. To implement these, we have made use of various Natural Language Processing (NLP) techniques along with the Wordnet knowledge base. We have tested the system with different corpus sizes and different coefficient of cosine similarity to explore this technique.

***Keywords*—**Question-Answering, Proximity, Semantic Similarity, Natural Language Processing and Synonyms.

I. INTRODUCTION

The Internet has proved to be a huge boon to people all over the world. When we want to find out about something or gather information about something, the best and quickest way to go about doing it would be to look it up online. The internet houses vast amounts of data, bigger than any library in the world, constantly updating itself everyday with new information and all this info is readily available to most of us at the tip of our fingers. As users struggle to navigate the wealth of online information now available, the need for automated question answering system becomes more urgent. Towards that end, many Search-Engines/Question-Answering Systems have been created over the past decade or so, that can return ranked lists of documents. For e.g. Google.com, Yahoo.com, Bing.com etc. Question-Answering systems and Search Engines are rarely ever able to give a correct and definitive answer to a question asked by a user. These systems merely do the task of providing links to pages of data that the user manually goes through, on his own, to find an answer that satisfies his/her search. Sometimes these links may or may not contain what the user is looking for, prolonging his search. This is especially deterring and time consuming for an individual. The best systems at present are now able to answer more than two thirds of factual questions in this context. However, sometimes, what the user is looking for could be like a needle in a haystack. People like to write huge responses on blogs or informational websites, which can be extra burden to find the answer. The answer the user wants, can be a single sentence or one that can go on for multiple sentences. While, factual answers can be easy to search today, the lengthy and non-factual answers are challenging.

Most of the time, the user browses many such domains that come up at the top of their search and which through no fault of their own, may not even contain the correct answer. Even if the user is looking at the right page, he has to read the document on his own for an answer. Furthermore, the answer to the user's question has every possibility of being updated or changed theoretically at any point in time. However, these changes do not show up on web pages very soon. Also, not all web pages update their answer. Many even leave the web page with the wrong answer! Pages that contain the outdated answer keep showing up on their search and so, the user has to go through more and more web pages to find the right answer.

In our system, we make use of a corpus comprising of news articles, across topics such as Sports, Entertainment, Business and Science and Technology from the archives of 'THE HINDU' newspaper's website [1]. We decided to opt for news articles for their informational content and because news is updated every day. People write new articles on daily basis and they make optimal use of the English vocabulary. In order to optimize the question answering with respect to time and memory, we pre-process the corpus. Here, the entire corpus is tokenized and undergoes stop word removal. Tokenization involves splitting up of text into units or tokens. The stream of characters in a natural language text must be broken up into distinct meaningful units (or tokens) before any language processing beyond the character level can be performed. Stop-word removal is the elimination of the set of frequently occurring words in the English language such as {the, is, in, has, have, etc.}, which very rarely contribute towards the meaning/semantics of the sentence and often act according to the role they serve i.e. as syntactic structures. Preprocessing the corpus allows for faster extraction of answers in later stages.

Document retrieval can involve any well-established technique available. For our system, we use the term frequency – inverse document frequency (tf-idf) method to extract documents. Here, an inverted index of the documents in the corpus needs to be created, which is used to extract relevant documents based on the user's question.

The Wordnet lexicon consists of groups of words, in hierarchical structure, that are synonyms and have relational pointers, such as "ISA" relation[4] or the hyponymy relation, to other synsets within the structure. Further, wordnet makes use of a concept called Lemma[1][2][3], which deals with the base form of words. Using these synonym sets and their lemma form, we are able to search for an even larger target set of textual information. Using the proximity scoring and sentence similarity technique, our system does the following important steps:

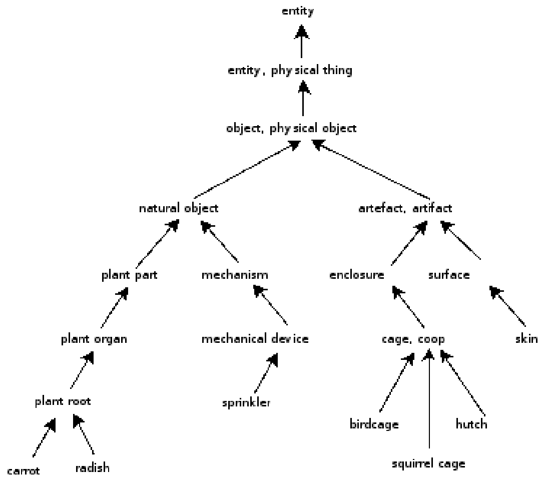


Figure 1. Wordnet Structure

i. Identify key words in the question asked.
ii. Make use of keywords, to identify passages in text documents that contain relevant information.
iii. Passages with Proximity score equal to or higher than the average Proximity score are forwarded to check for semantic similarity.
iv. The passage is checked for context wise similarity against the user question.
v. Passages getting a score equal to or higher than the average are summarized into an answer for the user.

Our system is able to search through articles/documents and provide to the user, a concise answer to his question. It does the searching and processing of answer for the user, who would instead have been searching through the documents on his own, manually. The user avoids the hassle of looking at one web page after another for an answer.

Rest of the paper is organized as follows, Section I contains the introduction of Concept of Question Answering using proximity scoring and Semantic Similarity , Section II contain the related work with regards to information retrieval using various means, Section III explains the Proximity and Semantic Similarity methodology in a stepwise manner with flow chart, Section IV describes results observed and discussions on the results and Section V concludes research work with future directions.

II. RELATED WORK

Most attempts in information retrieval is based on retrieving factual content. Search engines today are optimized in retrieving factual answers from the best-ranked web pages. Researchers are still finding ways to retrieve non-factual complex answers, attempting to model human reasoning. Some of the known existing system are:

i. **Standalone Encyclopedia software:** These were the earlier forms of getting information related to the desired topics. The user would type in the topic and the software would give information stored in its knowledge base. The information displayed is decided by the author which is already set. It would have the same information for a given keyword.
ii. **Search Engines:** First step towards getting answers is search engines. Answering method followed in this system is based on links to popular searches. Since the answers provided by existing search engines is biased i.e. content displayed is drawn up based on users browsing history and ranks the pages by the factor of popularity.
iii. **Phone Assistants:** Another widely used feature in the Smartphone is phone assistant. It uses voice recognition and obtain commands from the user. It uses specific voice commands and makes use of the default browser installed on the Smartphone and hands over the control to the browser.

Our technique is related to the computation of similarity between short sentences (Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett 2006), where a lexical database is used with the text similarity method to compare two short sentence. In our paper, we have tried to find the similarity between the important points of the question and the points mentioned in various passages extracted from documents and mention its results and drawbacks. We aim to show how a Proximity scoring [3] technique, used to retrieve documents based on their passages, can be used to effectively extract only relevant passages out of such documents in order to make the search process more efficient.

III. Methodology

As a preliminary step, the Question is preprocessed too. The question is pre-processed before any of the search related steps begin, in order to make it easier for the system to get to the answer/reply, and to allow the answer to be as detailed as possible without any distracters. These documents are pooled in a single file, in the order in which they are retrieved. The reason for this is, the information online is volatile. It would not suit our interests if we only take into account the latest information, or the outdated information. In order to account for such changes, taking place across days, we pool the textual information from the retrieved documents into a single file. This allows data across days to be processed together, incorporating the timeline as well. Furthermore, sometimes articles or web pages are not substantial enough to extract passages, although they may contain relevant information. Pooling the information makes up for this anomaly.

We then retrieve answers from these documents. To do this we perform the following steps:

A. Passage Retrieval

1) Identification of Passages

In order to process the pooled textual information for answers, we subdivide the information into “passages” of say 200-250 words each. These passages are created by using the presence of the keyword or one of its synonyms as a focal point. The number of passages extracted depends on the amount of information contained in the documents returned and the relevance of this content with the question asked, i.e. the presence of those important words and/or their synonyms, extracted from the user question. This ensures the systems looks at the right place within all the information available.

The concept of passage as a unit was used to retrieve documents by Man-Hung Jong, Chong-Han Ri, Hyok-Chol Choe, Chol-Jun Hwang in [3]. However, their approach only looked at the face value of query terms and not the semantics. Incorporating semantics into this technique, we extract passages as summarized below:

i. Identify Query terms from the question and their synonyms using Wordnet as the keywords.

ii. Within the documents retrieved, we identify the positions of every keyword.

iii. [3]Based on the positions identified in a document, a passage is extracted by taking into account the line containing the position and subsequent words and sentences that make an approximate 200-250-word strong passage.

A number of passages can be extracted in this manner. Some passages may contain sentences that are present in other passages as well. This happens because we take the position of the query word or its synonyms and extract a passage around it. So say if two such positions are close by (say < 100 words), but in two separate sentences, then there are chances that their respective sentences may appear in each other’s passages or that the passages may be the same. This ensures that no point of interest is omitted while trying to find an answer. If a passage misses a point that happens to be few words away, a subsequent passage will pick it up.

B. Proximity Score

To calculate the proximity score of a passage, we make use of the formula [3] shown below.

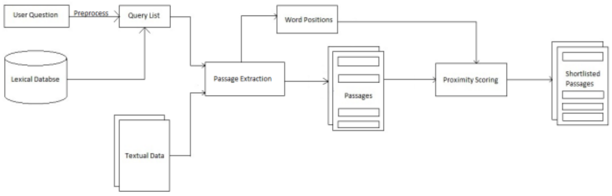


Figure 2 Passage Extraction and Proximity Scoring

$$Proximity\,Score\,(p1,p2) = \frac{1}{(1+(s*ln(1+dist(p1,p2))))} \quad (1)$$

Where,

p1 - position of word 1.

p2 – position of word2.

s – parameter that represents importance of the distance.

dist() – function that returns the distance between the two words.

The parameter ***s*** here will always be equal to 1, giving the distance between two words highest importance in the range [0, 1].

$$S(\omega) = \sum_{i=1}^{x}\sum_{j=1}^{y} Proximity\,Score(pi,pj) \quad (2)$$

Where,

ω : represents the passage under consideration.

i, j : represents the ***i***th and ***j***th word positions within a passage.

The total proximity score [3] of a passage is calculated as shown above[3].

$$\sum_{i=1}^{n} S_i(\omega) \div n \tag{3}$$

The Average Proximity score across all passages serves as the threshold while considering any of these passages for semantic similarity. This threshold is introduced to eliminate noise coming from passages whose proximity score is very low. These low scoring passage contain very less informational content.

C. Semantic Similarity

In order to calculate the similarity between the question and the passages extracted, we looked at the approach in [1]. The difference being that [1] confined itself to short sentences, whereas, our system, would compare a question from the user, i.e. a short sentence, and the passages retrieved through proximity scoring, i.e. long sentences or a combination of short sentences.

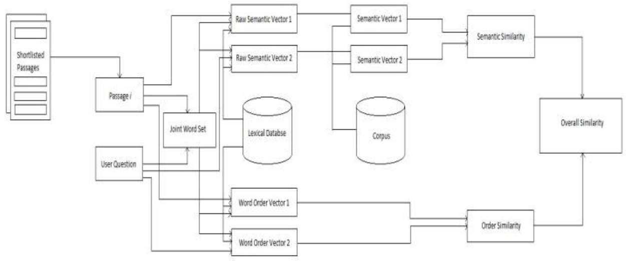


Figure 3. Semantic Similarity Calculation

We consider each passage separately against the question statement and this time, we include the stop words, as we aim to study the syntactic structures as well as semantics. A joint word set, comprising of unique words, is formed every time a new passage is brought up, so the similarity between the question and each of the passages is calculated separately. Hence, the order of the semantic and word order vector changes every time. The use of Wordnet knowledge base as in [1] is replicated here as well. This is due to the available hierarchical structure modeling the human common sense knowledge. In the hierarchy, more general semantics occur higher up, as compared to the specific semantics as you go lower down. This hierarchy is explored here, to identify semantics between the question and possible answers to these questions.

1) Semantic Vector

The steps towards calculating semantic vector remain almost the same as mentioned in [1], as follows:

- Let each individual word from both the Question and the Passage be part of their word sets, i.e. Q = $\{q_1,q_2\ldots..q_n\}$ and P = $\{p_1,p_2\ldots\ldots p_m\}$.
- Form a joint word set consisting of unique words from both Q and P, i.e. J.W.S = $\{w_1,w_2\ldots\ldots w_x\}$. The value of x will be different for different passages, as the value of m will be different.
- If w_i (i=0,1,2…..n, where n is length of J.W.S.) appears in the set P, i.e. in the question, then $\check{s}_i = 1$, where i indicates the position in the similarity vector s.
- If w_i does not appear in the set P, a semantic score between w_i and every word in the set P is calculated, as shown in [1]:

$$\check{s}(w_1,w_2) = e^{-\alpha l}.\frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \tag{4}$$

Where l is the shortest distance between the two words in the Wordnet hierarchy and h is the depth of the subsumer of the two words from the root word. l = 0, if the words belong to the same synset. l = 1, if the synsets for the two words contain common words, indicating partial similarity. However, the value of l is calculated by traversing the wordnet if the two words are neither in the same synset nor contain common words in ther respective synsets. w_1 = w_i and w_2 ϵ P. If $\check{s}_i < 0.05$, then $\check{s}_i$ is set to 0. The threshold of 0.05 is to reduce noise from the obvious dissimilar words. The values of α and β are set to 0.2 and 0.45 as is done in [1]. The highest semantic score between w_i and the word in set P is taken as the score of $\check{s}_i$.

- The raw semantic vector needs to be supplemented with the weight of the words in the their sources so that their importance is emphasized.

$$I(w) = 1 - \frac{log(n+1)}{log(N+1)} \tag{5}$$

Where, n is the frequency of the word and N is the total number of words in the respective sources. I ϵ [0,1]

- The similarity score between w_i and the set P is calculated for the similarity vector as:

$$s_i = \check{s}_i.I(w_i).I(\acute{w}) \tag{8}$$

Where $\check{s}_i$ is the similarity score calculated for w_i and is associated with the word $\acute{w}$ in the passage. This gives the final semantic vector.

- Similarly, the similarity vector for the question is also calculated using the set Qs.
- The semantic similarity score between the question and the passage is calculated using the two semantic similarity vectors as follows:

$$s_s = \frac{s_1.s_2}{||s_1||.||s_2||} \tag{7}$$

2) Word Order Vector

The word order vector adds structural information to this technique. Every word in a sentence or a passage contributes to the meaning in its own way. Hence, adding its structural information is essential. To incorporate the word order in the semantic similarity vector, we use the ordering of each word in the joint word set. With the first word of the joint word set numbered 1, the second number 2 and so on, the index of the words make up the values of the word order vector, ***r***, for both set Q and P. Just like the semantic vector, we follow the technique and conventions give in [1], as follows:

- If word w_i in J.W.S. is present in set P, ẃ, then the index of the word w_i in P is set as order for word w_i in the vector $\boldsymbol{r_i}$.
- If word w_i is not present in set P, then the index of the most similar word, ẃ, with a similarity score greater than a threshold of 0.05, is taken as a value for w_i in the vector $\boldsymbol{r_i}$.
- If the word w_i is not present in set P and nor does it have similarity with any word in set P, then the value for w_i in the vector $\boldsymbol{r_i}$ is 0.
- Having obtained the word order vector for the question set Q and the passage set P, we calculate the overall word order vector for passage P using,

$$\boldsymbol{S_r = 1 - \frac{||r_1 - r_2||}{||r_1 + r_2||}} \qquad \textbf{(8)}$$

Where,
$\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are the word order vectors for set P and Q.
$\boldsymbol{S_r}$ is the overall word order vector for the passage P, obtained by normalizing difference in word-order.

We can calculate the overall similarity score between the Question and the Passage using,

$$\boldsymbol{S(Q,P) = \delta S_s + (1-\delta)S_r} \qquad \textbf{(9)}$$

Where,
δ is 0.85 as set in [1]. δ decides the contributions of semantic and word order vector to the overall similarity between a passage and the question. Passages scoring above average are then considered for the answer.

$$\sum_{i=0}^{n} S_i \div n$$

IV. Results and Discussion

Our system is designed to pre-process the corpus once it is updated. Here, each document is tokenized, stop word removed and indexed using tf-idf (term frequency – inverse document frequency). This approach allows us to regulate the number of documents retrieved using cosine similarity(0.6-1). As the cosine similarity approaches 1, the search on pre-processed corpus is tightened, as the systems looks for words specific to those asked in the question. Since our project was not centred on document retrieval, we do not focus on this aspect of the project.

The cosine similarity value used, affects the number of documents retrieved by the system and hence, the average similarity score of the passages. As the cosine similarity score reached 1, less documents were retrieved. This was because the system would look for the very specific words mentioned in the question. If the user asked "what were maruti suzuki's sales in December?", the system would specifically look for the words maruti, Suzuki, sales and December. With a lower cosine similarity score, the system would not just look for documents containing the words mentioned in the question, but words closely associated to them as well, such as car, month, sold, selling etc.. This would produce larger number of documents, as illustrated in **Figure 4** below.

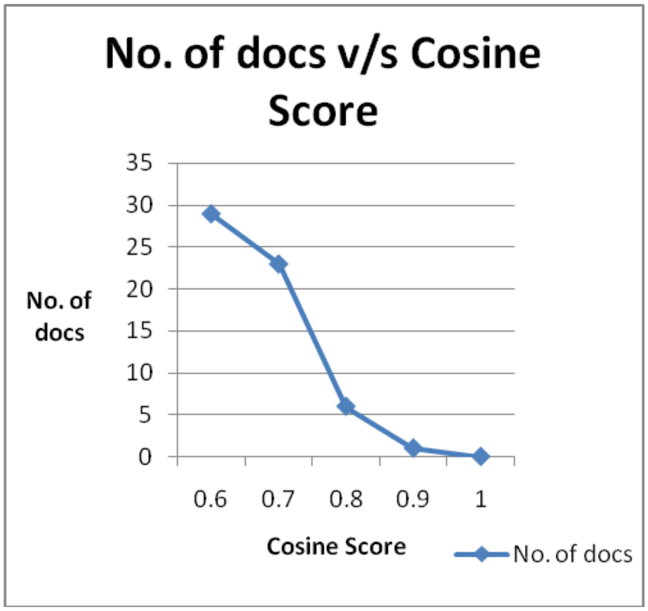


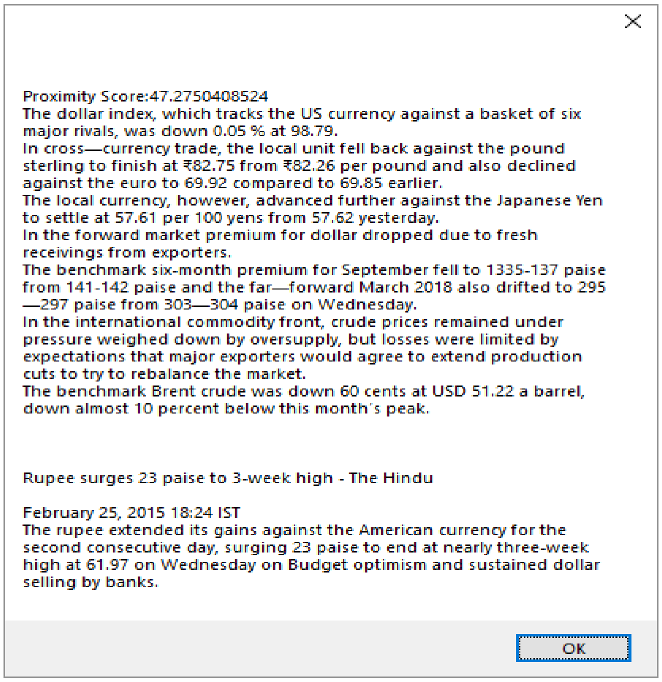
Figure 4 Number of Documents Retrieved V/S Cosine Similarity Threshold

Here, we asked the same query," Samsung launches galaxy s8", with varying values of cosine similarity ranging from 0.6-1. The number of documents retrieved reduced as the cosine similarity threshold got closer to 1. For research purposes, we maintained the corpus size to be just 500 documents. However, similar behaviour was reciprocated at larger corpus sizes as well.

A. Extracting Passages

The Proximity Scores of each passage varied with every individual document due to the way each of the keywords or their synonyms spread within the document. Each such word including its synonyms, serving as the focal point would return a passage. Sometimes a passage extracted around a focal point would be very similar, if not the same, to another passage, extracted around another word, not too far away from the former. For the query “Dollar versus Rupee”, the system produced passages as follows:

The extraction process gave an average proximity score of **5.6650564839** from 119 documents searched, with scores ranging from **47.2750408524** to **0** as shown below. Passages that score less than the average, obviously spoke less of the topic at hand as shown. Hence, considering passages with proximity score higher than the average was sensible.

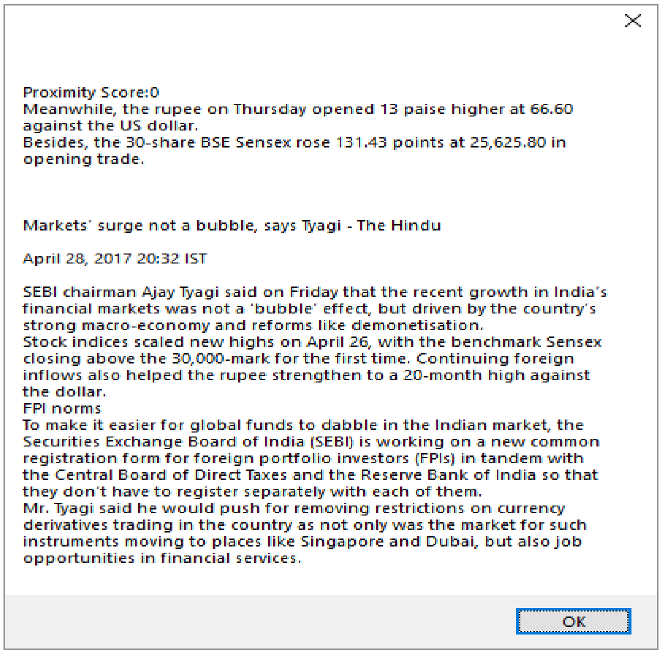


Proximity Score:47.2750408524
The dollar index, which tracks the US currency against a basket of six major rivals, was down 0.05 % at 98.79.
In cross—currency trade, the local unit fell back against the pound sterling to finish at ₹82.75 from ₹82.26 per pound and also declined against the euro to 69.92 compared to 69.85 earlier.
The local currency, however, advanced further against the Japanese Yen to settle at 57.61 per 100 yens from 57.62 yesterday.
In the forward market premium for dollar dropped due to fresh receivings from exporters.
The benchmark six-month premium for September fell to 1335-137 paise from 141-142 paise and the far—forward March 2018 also drifted to 295—297 paise from 303—304 paise on Wednesday.
In the international commodity front, crude prices remained under pressure weighed down by oversupply, but losses were limited by expectations that major exporters would agree to extend production cuts to try to rebalance the market.
The benchmark Brent crude was down 60 cents at USD 51.22 a barrel, down almost 10 percent below this month's peak.

Rupee surges 23 paise to 3-week high - The Hindu

February 25, 2015 18:24 IST
The rupee extended its gains against the American currency for the second consecutive day, surging 23 paise to end at nearly three-week high at 61.97 on Wednesday on Budget optimism and sustained dollar selling by banks.

Figure 5. Example: Max Proximity Score



Proximity Score:0
Meanwhile, the rupee on Thursday opened 13 paise higher at 66.60 against the US dollar.
Besides, the 30-share BSE Sensex rose 131.43 points at 25,625.80 in opening trade.

Markets' surge not a bubble, says Tyagi - The Hindu

April 28, 2017 20:32 IST

SEBI chairman Ajay Tyagi said on Friday that the recent growth in India's financial markets was not a 'bubble' effect, but driven by the country's strong macro-economy and reforms like demonetisation.
Stock indices scaled new highs on April 26, with the benchmark Sensex closing above the 30,000-mark for the first time. Continuing foreign inflows also helped the rupee strengthen to a 20-month high against the dollar.
FPI norms
To make it easier for global funds to dabble in the Indian market, the Securities Exchange Board of India (SEBI) is working on a new common registration form for foreign portfolio investors (FPIs) in tandem with the Central Board of Direct Taxes and the Reserve Bank of India so that they don't have to register separately with each of them.
Mr. Tyagi said he would push for removing restrictions on currency derivatives trading in the country as not only was the market for such instruments moving to places like Singapore and Dubai, but also job opportunities in financial services.

Figure 6. Example: Min Proximity Score

B. Semantic Similarity Scoring

Once proximity scores were established, it was imperative to determine their similarity with the question in order to determine the answer. The semantic similarity of the passages with the question asked, showed exactly how similar a passage was to the question statement. The similarity between every word in the passage and every word in the question including their synonyms, for every passage, showed a semantic similarity score in the range [0,1]. This was consistent with the cosine similarity parameters that state, cosine (0^o) = 1 (Exactly Similar) and cosine (90^o) = 0 (Completely Dissimilar).

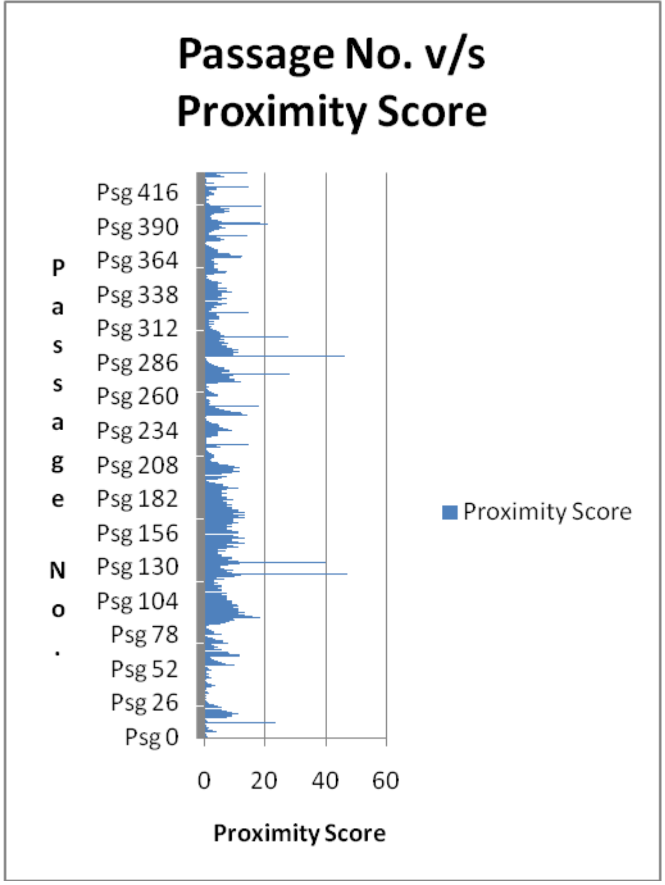
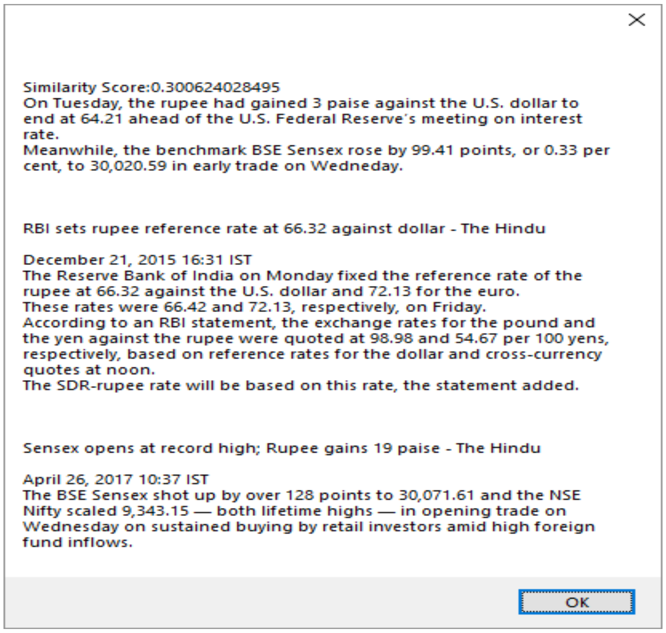


Figure 7. Proximity Score Chart

In the illustrations provided, most passages averaged a similarity score of **0.232515939783**, with passages ranging in semantic similarity score between **0.300624028495** and **0.191711290599** as shown below.

The similarity scores of passages extracted through proximity scoring look as shown below. These passages are the most similar in semantics with the question, from among 119 passages that were searched initially by the question answering system. As explained in [1], most similarity scores will be very low as we calculate the similarity of each individual words in the question and the passages. Hence, most words would be dissimilar from most others, except a few. Hence, the purpose of the threshold while entering semantics similarity values in the vectors. We need to avoid such noise as much as possible.

Although “Dollar versus Rupee” is a very trivial/factual question, we aim at showing the “Why?” of the question, and the results are as shown below.


Similarity Score:0.300624028495
On Tuesday, the rupee had gained 3 paise against the U.S. dollar to end at 64.21 ahead of the U.S. Federal Reserve’s meeting on interest rate.
Meanwhile, the benchmark BSE Sensex rose by 99.41 points, or 0.33 per cent, to 30,020.59 in early trade on Wedneday.

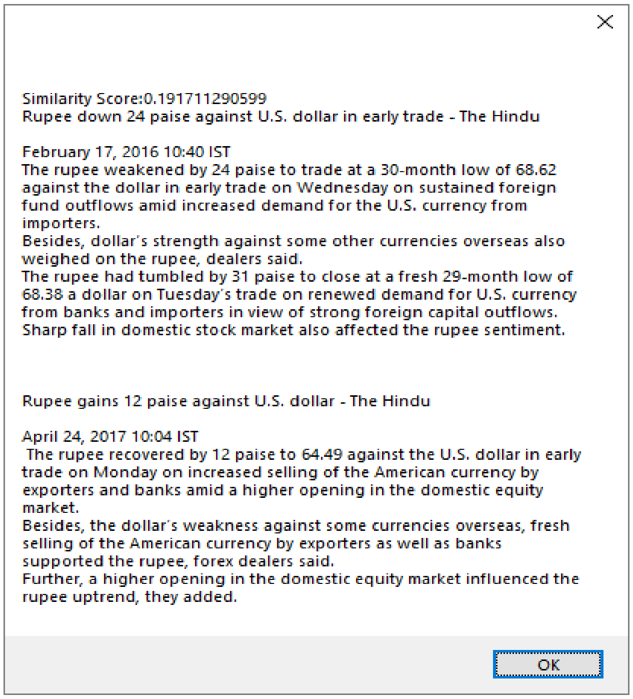
RBI sets rupee reference rate at 66.32 against dollar - The Hindu

December 21, 2015 16:31 IST
The Reserve Bank of India on Monday fixed the reference rate of the rupee at 66.32 against the U.S. dollar and 72.13 for the euro.
These rates were 66.42 and 72.13, respectively, on Friday.
According to an RBI statement, the exchange rates for the pound and the yen against the rupee were quoted at 98.98 and 54.67 per 100 yens, respectively, based on reference rates for the dollar and cross-currency quotes at noon.
The SDR-rupee rate will be based on this rate, the statement added.

Sensex opens at record high; Rupee gains 19 paise - The Hindu

April 26, 2017 10:37 IST
The BSE Sensex shot up by over 128 points to 30,071.61 and the NSE Nifty scaled 9,343.15 — both lifetime highs — in opening trade on Wednesday on sustained buying by retail investors amid high foreign fund inflows.

Figure 8. Example: Max Semantic Similarity Score


Similarity Score:0.191711290599
Rupee down 24 paise against U.S. dollar in early trade - The Hindu

February 17, 2016 10:40 IST
The rupee weakened by 24 paise to trade at a 30-month low of 68.62 against the dollar in early trade on Wednesday on sustained foreign fund outflows amid increased demand for the U.S. currency from importers.
Besides, dollar’s strength against some other currencies overseas also weighed on the rupee, dealers said.
The rupee had tumbled by 31 paise to close at a fresh 29-month low of 68.38 a dollar on Tuesday’s trade on renewed demand for U.S. currency from banks and importers in view of strong foreign capital outflows.
Sharp fall in domestic stock market also affected the rupee sentiment.

Rupee gains 12 paise against U.S. dollar - The Hindu

April 24, 2017 10:04 IST
The rupee recovered by 12 paise to 64.49 against the U.S. dollar in early trade on Monday on increased selling of the American currency by exporters and banks amid a higher opening in the domestic equity market.
Besides, the dollar’s weakness against some currencies overseas, fresh selling of the American currency by exporters as well as banks supported the rupee, forex dealers said.
Further, a higher opening in the domestic equity market influenced the rupee uptrend, they added.

Figure 9. Example: Min Semantic Similarity Score

The final answer displayed by our system was a summary of all the textual information that clear the semantic similarity threshold. For our system we made use of the Gensim summarizer[6] and it showed the following result.

Here, some of the information has duplicated itself, due some passages containing identical information. However, human standards show the displayed answer is semantically accurate.

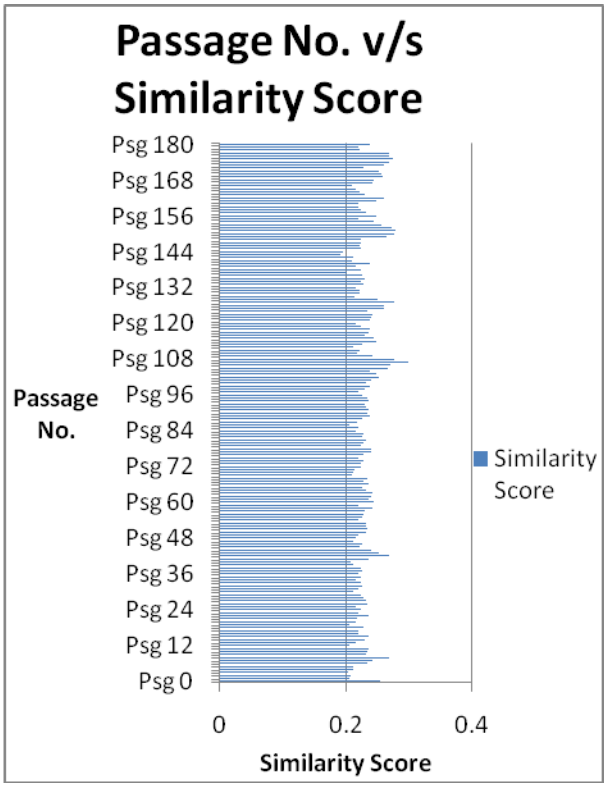
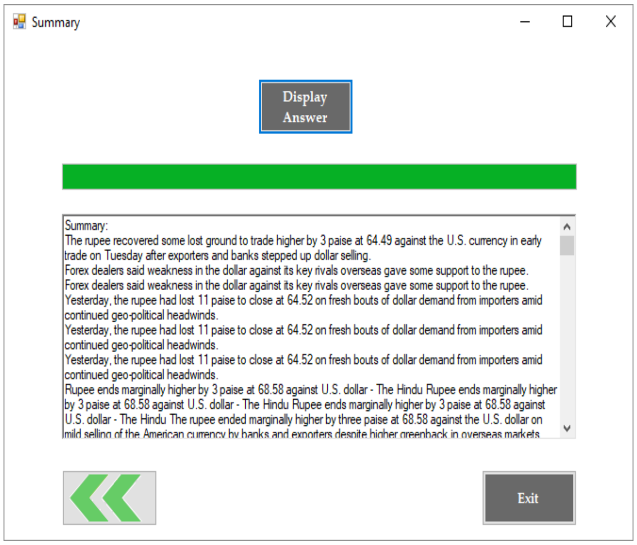


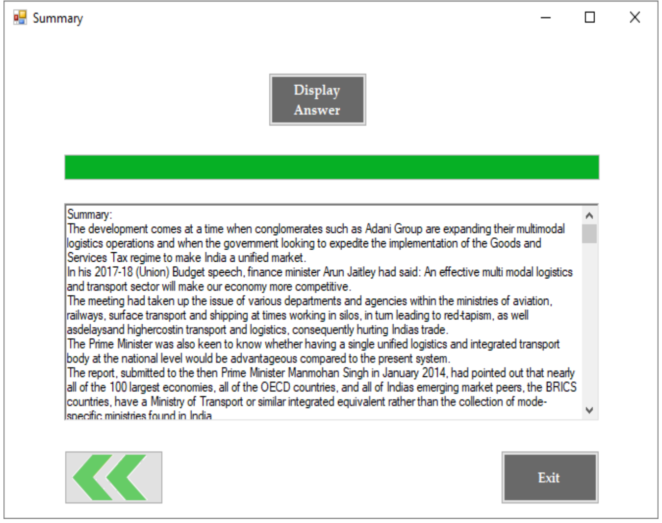
Figure 10. Example: Semantic Similarity Score Chart



Summary:
The rupee recovered some lost ground to trade higher by 3 paise at 64.49 against the U.S. currency in early trade on Tuesday after exporters and banks stepped up dollar selling.
Forex dealers said weakness in the dollar against its key rivals overseas gave some support to the rupee.
Forex dealers said weakness in the dollar against its key rivals overseas gave some support to the rupee.
Yesterday, the rupee had lost 11 paise to close at 64.52 on fresh bouts of dollar demand from importers amid continued geo-political headwinds.
Yesterday, the rupee had lost 11 paise to close at 64.52 on fresh bouts of dollar demand from importers amid continued geo-political headwinds.
Yesterday, the rupee had lost 11 paise to close at 64.52 on fresh bouts of dollar demand from importers amid continued geo-political headwinds.
Rupee ends marginally higher by 3 paise at 68.58 against U.S. dollar - The Hindu Rupee ends marginally higher by 3 paise at 68.58 against U.S. dollar - The Hindu Rupee ends marginally higher by 3 paise at 68.58 against U.S. dollar - The Hindu The rupee ended marginally higher by three paise at 68.58 against the U.S. dollar on mild selling of the American currency by banks and exporters despite higher greenback in overseas markets

Figure 11. Answer displayed to User

On an experimental note, we queried our system with a general question such as "Prime Minister of India". The system was able to retrieve 243 documents bearing relevance to the Prime Minister of India, out of a total corpus of 4000 documents. Giving an average similarity score of 0.392761836947, the answer provided spoke across all categories.



Summary:
The development comes at a time when conglomerates such as Adani Group are expanding their multimodal logistics operations and when the govemment looking to expedite the implementation of the Goods and Services Tax regime to make India a unified market.
In his 2017-18 (Union) Budget speech, finance minister Arun Jaitley had said: An effective multi modal logistics and transport sector will make our economy more competitive.
The meeting had taken up the issue of various departments and agencies within the ministries of aviation, railways, surface transport and shipping at times working in silos, in tum leading to red-tapism, as well asdelaysand highercostin transport and logistics, consequently hurting Indias trade.
The Prime Minister was also keen to know whether having a single unified logistics and integrated transport body at the national level would be advantageous compared to the present system.
The report, submitted to the then Prime Minister Manmohan Singh in January 2014, had pointed out that nearly all of the 100 largest economies, all of the OECD countries, and all of Indias emerging market peers, the BRICS countries, have a Ministry of Transport or similar integrated equivalent rather than the collection of mode-specific ministries found in India

Figure 12. General Query Example

V. Conclusion and Future Scope

The Question Answering system using Proximity Scoring and Semantic Similarity was successfully implemented. The system was able to isolate the important passages, based on the question asked by the user, from the pre-processed corpus. The passages isolated, contained information relevant to the question asked and were able to answer significantly. As per the design, the system not only isolated passages that contained the exact words mentioned in the question, but also isolated those that contained words closely associated to the question words in terms of semantics. The proximity scoring algorithm was able to extract passages making sure they were not too big and neither too small. At the same time, it made sure most major points from a document were extracted.

The similarity scores for each of the extracted passages when calculated did not vary much, with scores differing in the degree of 0.01 to 0.00001. This was because of the semantics part of the Proximity scoring algorithm. Hence, taking information from passages with a similarity score above the average score provided significant answers to the question asked. Queries where few documents were retrieved, sometimes showed higher semantic similarity value for passages, then those where more number of passages were retrieved. This shows, that the search result gives importance to the similarity in semantics to the question asked, rather than just retrieving unrelated answers.

FUTURE SCOPE

In this model, the semantics is checked using the Wordnet lexicon. Hence, this approach works with information that comes with the downloaded knowledge base. However, the English language is constantly updating itself with new words in its vocabulary. Incorporating such a function that can keep up with the changes in the English language is something that can be added in the future.

While implementing the system, we had already downloaded and pre-processed our corpus. This was due to limitations in processing power it would require in order to obtain relative data and pre-process it dynamically, before going ahead with the search process. Although we did make use of a web crawler to get the information, simply obtaining the data and pre-processing it at the same time would be a tedious task. Hence, an application that could do such a thing could be one for the future.

Our corpus comprised of articles from the archives of 'THE HINDU' newspaper's website. This served to be a reliable source in terms of information content. However, all information cannot be necessarily obtained from such articles alone. Hence, a dynamic search engine, which could search the relative article from the internet and extract it for pre-processing could be done in the future.

Although our system gave results consistent with the requirements and also the human standard of evaluation, there is no definite benchmark to check the results with. This is mainly due to the large amount of parameters and inferential techniques that would be required to implement such a thing.

The system gave an answer that still contained sentences that duplicated in the answer. Although this resulted due to part of our processing, eliminating such discrepancies can be part of the future.

VI. References

[1] Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett, "Sentence Similarity Based on Semantic Nets and Corpus Statistics", in IEEE Transactions on Knowledge and Data Engineering, VOL. 18, NO. 8, AUGUST 2006.

[2] Daniel Jurafsky & James H. Martin, "Speech and Language Processing",

[3] Man-Hung Jong, Chong-Han Ri, Hyok-Chol Choe, Chol-Jun Hwang, A Method of Passage-Based Document Retrieval in Question Answering System. https://arxiv.org/ftp/arxiv/papers/1512/1512.05437.pdf.

[4] Apra Mishra, Santosh Vishwakarma, Analysis of TF-IDF Model and its Variant for Document Retrieval, ,2015.

[5] Vicedo, Jose Luis & Ferrández, Antonio, A Semantic approach to Question Answering systems.

Authors Profile

Mr. Shaunak S. Phaldessai pursed Bachelor of Engineering in Computer Engineering from Don Bosco College of Engineering, Goa University, India in the year 2017. Thereafter, he has worked as a Software Developer for about a year. Curently, he is aspiring to pursue Masters in Data Science.

Mr. Amey D. S. Kerkar pursed Bachelor of Engineering in Computer Engineering and Master of Engineering in Information Technology from Goa University, India in year 2006 and 2011 respectively. He is currently working as Assistant Professor in Department of Computer Engineering, Don Bosco College of Engineering, Goa, India since 2013. He has over 10 years of teaching experience in Engineering. He has published research papers in reputed international journals. His recent work is in the field of Opinion Mining Using NLP. His main research work focuses on NLP, Data Mining ,Cryptography and Automata Theory. He has 5 years of research experience.