INSIGHTICA INC.

Scope of the Work

Insightica Text Analysis Engine

5/27/2015

Table of Contents

Executive Summary

This RFP is intended to provide the scope of the work for a text analysis engine. The intended audience are any interested individuals or companies that are willing to submit a proposal including all deliverables listed under the deliverables section of this RFP.

Please submit a customized answer to this RFP, including a customized proposal with break downs of milestones, price, and time.

In Scope

1. A cloud based software that detects specific words in a large text and runs set of rules that apply to that word,
2. UI to add rules to the rule engine,
3. Interface to upload a text file or receive a URL to a webpage that includes the text,
4. Get a URL for a webpage and capture the text on that webpage,
5. Third party API calls or use of third party SDKs as required,
6. Database structure to store raw and processed data,
7. Ability to upload multiple word lists from an excel file containing,
8. Ability to update (Add/Delete/Replace) a previously uploaded list of words,
9. Interface to present results with the ability to export to MS Excel.

Out of Scope

1. Although we require an interface for ease of use, a fully implemented UI/UX is out of scope,
2. High performance - ***Please see item 2 under the assumptions section***,
3. User authentication and user management.

Assumptions

1. The user interface will be simple and will be used by technical people during the proof of concept.
2. The proposal should consider the desire to scale the engine so that it can process large amount of text. However, the first version is intended to be used for small pilot. Scalability is out of scope for this phase but current implementation should not impede or constraint scalability in future.
3. The proposal is based on using key software architecture principles as well as best practices for design and implementation.

RFP Functional Deliverables

1. Recommended choice of technology and implementation platform/environment,
2. Recommended choice of best practices for **software architecture** design and implementation,

3. Proposed high logical view of the system components,
4. Implementation schedule including high level task breakdown and duration,
5. Cost proposal that includes hourly rates as well as a grand total to complete the project,
6. Proposed process to handle change request prior to hand off,
7. Warranty period,
8. Rates for post-production maintenance and enhancements,
9. System performance (text throughput – words per sec and number of concurrent users).

RFP Non-functional Deliverables

1. Brief summary of any similar work completed in the past year,
2. Company portfolio or CV for individual responders,
3. Terms and conditions.

High level Requirements and Acceptance Criteria

User interface for Text Loading

1. The interface will be web based,
2. The interface will have a field to upload a Text and MS Word file,
3. The interface will have a field to insert a URL to get the text from a webpage and all frames within that specific webpage.

Field ID	Field	Type	O/M/C *	Values	notes
1	File upload	browser address	C		one of the two fields for "file upload" or "URL" is mandatory
2	URL	Free form	C		one of the two fields for "file upload" or "URL" is mandatory
3	Description	Free form	O		
4	Save	button	N/A		Uploads the file/captures if any date in other fields
O/M/C * Optional/Mandatory/Conditional					

User Interface for Rule Loading

1. The interface will be web based,
2. The interface will receive a combination of rule attributes (up to 20 attributes per rule),
3. Rule attributes will be stored and then used to run text processing rules.
4. Following table provides the fields on the page:

Field ID	Field	Type	O/M/C *	Values	notes
1	Rule ID	Numeric	M		Unique, Incremental system generated, non-editable
2	Function	Checkbox	M	TBD	
3	Word	free form	O		
4	Multi Word Rule	Dropdown	O	Yes/No	
5	Forward Search	Numeric	O		
6	Word List	Dropdown	O	TBD	Shows list of existing word list/ assigns the rule to all words on that list
7	Word order	numeric	C		System generated to keep the order of the words as entered by user/Starts at 1 for each rule/ is Editable/can't have the same order for two words in the same rule (should be unique value in each rule)
8	POS	Dropdown	O	TBD	
9	Type	Dropdown	O	TBD	
10	Negation	Dropdown	O	Yes/No	
11	Add word	button	N/A		Regenerates fields 3 to 9 - Word order will go up one unit
12	Status	Dropdown	M	Active/Inactive	
13	Save	button	N/A		Saves the rule to the rule engine
O/M/C * Optional/Mandatory/Conditional					

Part of Speech (POS) Tagging

1. Part of Speech Tagging is a backed process that involves a third party API or SDK,
2. Each rule has an indicator for part of speech as mentioned previous section,
3. For specific part of speech selection, the application will call a third party API to identify POS,
4. Rule engine will use the POS for executing rules.

Example of a Part-Of-Speech (POS) Tagger:
http://nlp.stanford.edu/software/tagger.shtml

Word List Upload

1. Word list upload is done using a web UI,

2. The description of the list will be added to the “Word List” drop down on the “Rule Loading” interface.

Field ID	Field	Type	O/M/C *	Values	notes
1	File upload	browser address	C		
3	Description	Free form	O		Mandatory if a file is uploaded/ description to be added to the “Word List” drop down on the “Rule Loading” interface
4	Save	button	N/A		Uploads the file/captures if any date in other fields
O/M/C * Optional/Mandatory/Conditional					

Engine

- The rule engine will host all rules for the application,
- Rules are attributed to words,
- Each word will have specific set of rules,
- Rule engine will process the text uploaded using the Text Upload Interface,
- Rule engine will pick each word in the text and will execute all rules that are attributed to that word.

Rule Examples

Rule Example 1

1. If sentence starts with a “First person agent as a pronoun (FPAP)” from our specific list of “FPAPs” (for example “I”), and
2. If it is followed by a “Verb” (a part of speech tagger API or SDK will be used to identify the verb), and
3. If it is followed by a “Noun” from our specific list of “basketball” nouns,
4. Then it counts as one point for “basketball” category.

Use case:
Sentence “I play basketball.” confirms with the above rule and should counts as one point for “basketball” category.

Rule Example 2

1. If sentence has the specific word “player” (plural or singular), and

2. The sentence does not have the specific word “very”, and
3. The sentence is “negative”, and
4. The sentence has a “Noun” from our specific list of “basketball” nouns and ,
5. Then it counts as one point for “basketball” category.

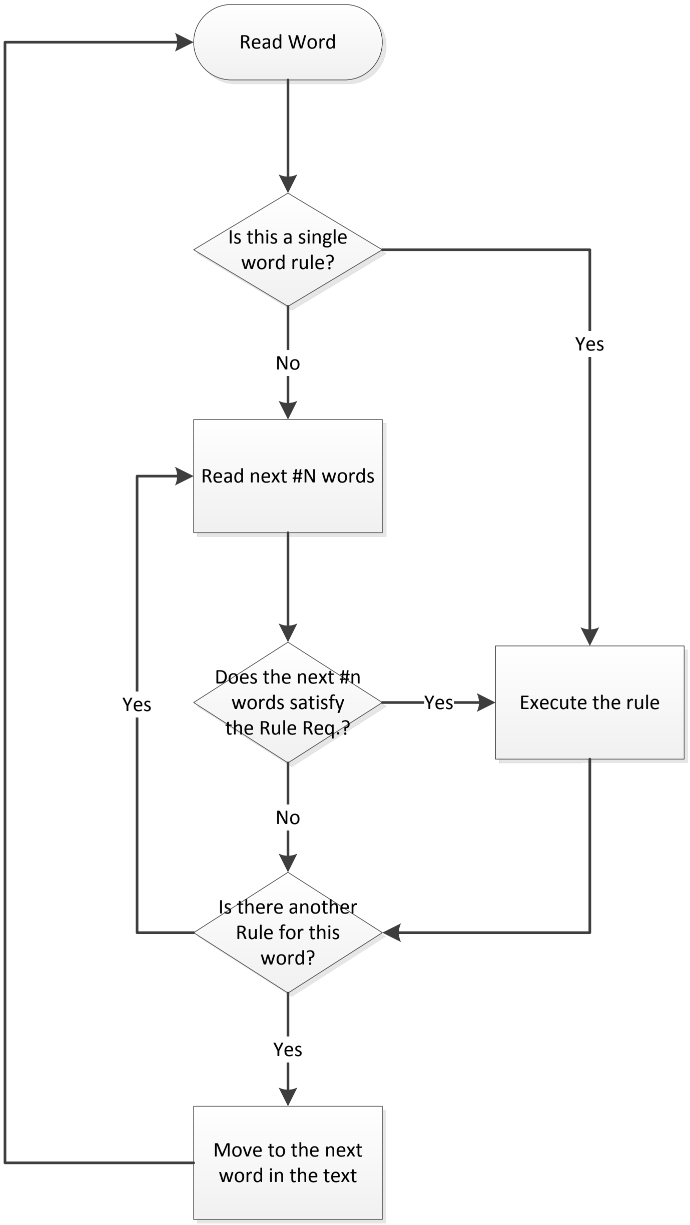
Use case:
Sentence “Basketball players are not short.” confirms with the above rule and should counts as one point for “basketball” category.

Engine Process Flow

Please note that in the flowchart below the “Read next #N words” includes an “OR” function in which whichever condition satisfies first would trigger the next step in the flow:

1. “Read the #N words”, OR
2. “Read words till you face any of these punctuation marks: Period “.” or Question mark “?”

Use case:
If “#N” is set as “10”, and the sentence is ““Basketball players are not short.”, then condition “2” becomes satisfied first as sentence has 5 words and a “Period .” is placed before reaching the 10th word.


Read Word
Is this a single word rule?
No
Yes
Read next #N words
Does the next #n words satisfy the Rule Req.?
Yes
Yes
Execute the rule
No
Is there another Rule for this word?
Yes
Move to the next word in the text

Output interface

1. Output is a web UI with a table in it,
2. Tables are sortable with column values,
3. The output can be exported and saved as excel.