Relevance and Support calculation for Health information

S. Suresh Kumar
Department of Information Technology
Assistant Professor
JNTU Hyderabad
Hyderabad
9440936885
sureshsanampudi@gmail.com

L Naveen
Department of Information Technology
Assistant Professor
BVRIT
Hyderabad

ABSTRACT
Consumer health information search (CHIS) is a forum of information retrieval that has organized two tasks to be performed. The first task includes the identification that whether a given query is relevant or irrelevant to the sentences available in the document. The second tasks talks about finding the nature of support of a sentence in the document to the query.

Task 1 i.e identification of relevance is done by performing different similarity measures calculations and finally averaging the obtained score to find the relevance nature of a sentence. The second task is solved by using a special type of support vector machine called C-support vector machines that can handle multiclass support. The results obtained from CHIS organizers shown that model developed for second task shown promising results than that of the model developed for the first task.

Keywords
C Support Vector Machines, TF-IDF, Jaccard coefficient, cosine similarity.

1. INTRODUCTION
In this work we explore the Consumer Health Information Search (CHIS) task for finding the relevant information for the user query from the given collection of sentence dataset. The task 1 implementation of CHIS aims to identify whether a given sentence is relevant to the query or not. Whereas task 2 aims at the identification of whether support nature of the sentence with respect to the query. In Task 1 of CHIS, retrieve the relevant information related to the user query, different similarity measures [5][6] have been used. The similarity coefficient was computed between the query and sentences given in the document collection. The average of these coefficients was identified and based on that value it is decided whether the sentence is relevant to query or not. Task 2 of CHIS aim is to identify whether each of the sentence in the given document collection is supporting or opposing or neutral to the claim made in the query. It was treated using a special type of support vector machine that includes the c-factor[7].

The rest of the paper is organized as follows. Section 2 discuss about various approaches used to find the relevance computation between the query and each sentence from document collection. Section 3 explain implementation of achieving the support nature of a sentence with respect to query. Section 4 elaborates the dataset description and queries used for search. Section 5 concludes the paper.

2. TASK-1 Relevance Identification
To retrieve the relevant collection of sentences to the query, we have calculated the similarity measures between the given query and the sentence collection. Similarity between query and sentence collection was computed both in syntactic and semantic aspect. The similarity measure reflects the degree of closeness or separation of the target objects. Choosing an appropriate similarity measure is also important for information retrieval task. In general, similarity measures plot the distance or similarity between the symbolic descriptions of two objects into a single numeric value[5]. Several syntactic similarity measures have been implemented in our model for task 1, few of them are: Euclidean distance, cosine similarity, jaccard coefficient.

In implementing CHIS task-1 we have used cosine similarity, jaccard coefficient, TF-IDF similarity to find the relevance with respect to syntactic nature of sentence and semantic similarity to find semantical relevance. The average score of the obtained scores were calculated to find the syntactic and semantic similarity of a given sentence with respect to that of query.

2.1 Cosine similarity
In this measure computation the sentences and query are represented as term vectors, the similarity is quantified as cosine angle between the query and a sentences vector that is, so-called cosine similarity. Cosine similarity is the most widespread similarity measures applied to check similarity between texts.

Given a Sentence collection (S) and query (Q), the similarity coefficient between them is computed using following formula:

$$SC1(\vec{S}, \vec{Q}) = \frac{\vec{S} \,.\, \vec{Q}}{|\vec{S}| \; X \; |\vec{Q}|}$$

Where $\vec{S}$ *and* $\vec{Q}$ are vector representation of sentence and query.

2.2 Jaccard Coefficient
The Jaccard coefficient, finds the Similarity measures between finite sample sets. It is defined as the cardinality of the intersection of sets divided by the cardinality of the union of the sample sets[3]. For text similarity jaccard coefficient compares the sum of weight of shared terms to the sum of weights terms that are present in either of the document but are not shared terms. The formal definition is:

$$SC2(\vec{S}, \vec{Q}) = \frac{\vec{S} \,.\, \vec{Q}}{|\vec{S}|^2 + |\vec{Q}|^2 - \vec{S} \,.\, \vec{Q}}$$

Where $\vec{S}$ *and* $\vec{Q}$ are vector representation of sentence and query.

2.3 TF-IDF Similarity
TF-IDF measures are a broad class of functions that are used for computing similarity and relevance between queries and documents. The basic idea is that, the more frequently a word appears in text, the more indicative that word is of the topicality of the text; and that the less frequently a word appears in a document collection, the greater its power to categorize between relevant or irrelevant.

The similarity function:

$$SC(S,Q) = \sum_{w \in Q \cap R} \log(tf_{w,Q} + 1) \log(tf_{w,S} + 1) \log\left(\frac{N+1}{df_w + 0.5}\right)$$

Where $tf_{w,Q}$ is the number of times word w appears in query sentence Q; $tf_{w,S}$ is the number of times word w appears in sentence S; N is the total number of sentences in the collection; and df_w is the number of sentences that w appears in.

2.4 Semantic Similarity

Semantic similarity measure the text similarity that is derived from semantic and syntactic information contained in the given texts. To compute the similarity, for each sentence, a raw semantic vector is derived with the help of lexical database; and also a word order vector is formed for each of the sentences using the same information from lexical database. Each word in a sentence contributes differently to the complete meaning of the whole sentence, the importance of a word is weighted by using information content resultant from corpus by combining the raw semantic with information from the corpus, a semantic vector is found for each of the two sentences[3]. Semantic similarity is computed based on the two semantic vectors. An order similarity is computed using two order vectors. Finally, the overall similarity is derived by combining order similarity and semantic similarity.

To find the relevance nature of the sentence to the given query, all these values of different similarity were averaged. A threshold was kept to say the sentences are relevant if they fall above these threshold value and irrelevant if they fall below the threshold.

3. Task 2 Support Calculation

With the explosive growth of the social media like twitter, face book, blogs and microblogging are used to search, extract information from these to help in decision making. As lots of information available from various sources and in diverse nature it becomes very difficult to identify whether the information is supporting or opposing or neutral to the user query

Identification of support of a sentence towards the query was recognized using a special class support vector machines that uses “C Factor” . Basically support vector machine is a binary classifier but to obtain the class of “neutral” we used a special category of support vector machine namely C-Support Vectors Classification which is based on libsvm[7].

In these support vector machine the first step is to find convert the collection of given sentences in a document into Term Frequency and Inverse Document Frequency (TF-IDF) feature vector. The next step identify whether the feature should be made of a word or with a character of n-gram. The lower and upper boundary of the range of n-values were extracted for different n-grams for all values of n such that n lies between min_n <= n <= max_n. After this step we need to build a vocabulary that consider top max_features ordered by term frequency across the corpus. Next a learning method has to be applied to learn vocabulary, IDF and return term document matrix.

We have used parameters C and penalty parameter of the error term. Kernel type is used along with Radial Basis function (RBF). When training SVM with RBF kernel, the parameters required is C. Lower the value of C makes the decision boundary smooth and higher the value of C makes classifying all the training examples correctly. C-SVM method provide a grid search method that implements a fit and score method that includes various function to be implemented such as probability prediction, decision functions and perform transformation and inverse transformation.

C-SVM is considered as supervised learning task, in which a model is built to learn from the training data and to predict the class label for the unseen data. Support Vectors constructs a hyperplane or set of hyperplanes in high dimensional space which is used for identification of support.

4. Implementation Model

For implementing CHIS task, the organizing committee has been given with training dataset of five documents in each approximately with 300 sentences and queries for retrieval process. Training dataset consisting of total of three attributes, in which attribute1 consists of sentence, attribute2 consists of relevant or irrelevant and attribute3 consists of polarity of the sentence towards the query as oppose, support or neutral.

The steps followed to complete the implementation of CHIS task1 is as follows, where the user query, training dataset and test dataset are taken as inputs. On the given inputs pre-processing steps has been applied. The pre-processing steps include stop word elimination and all the letters in the query and sentences were converted into lowercase before performing actual tasks of CHIS.

The Relevance identification task is to find whether a given sentence is relevant/irrelevant to the query. To achieve this task, similarity measure between the given query and each sentence from the document collection is computed using each of the techniques namely cosine similarity, jaccard coefficient, TF-IDF similarity and semantic similarity shown in the diagram. The overall similarity is considered as the average of the above all similarity measures. After computing the overall similarity between each pair of query and sentence from the document collection, if the similarity measure exceeds the threshold value, then the sentence in the document collection is considered as relevant else it is considered as irrelevant.

After evaluating the similarity measure between query and each pair of sentences in the document collection, the training dataset is used to train by the C-Support Vector Machine (SVM) classifier to predict the class label for the test data. Actually the normal SVM classier classifies only positive and negative but in order to identify the neutral nature of a given sentence we a used a special “C factor” in the SVM that identifies a marginal values between upper and lower planes that used TF-IDF feature as a measure.

5. Conclusion

Consumer health information search provide two tasks. Task 1 is about the identification of the relevant nature of the sentence with that of the query. Task 2 is about the identification of support (positive/negative/neutral) of the sentence with respect to the query. A framework has been designed to achieve this tasks. To achieve task 1 we have computed several similarity measures to find syntactic and semantic similarity between the sentence and the query. Task 2 is a support calculation of a sentence towards the given query. To achieve this a special type of C-support vector classification is used. It uses a TF_IDF feature and incorporate the n gram approach to learn the vocabulary. Using these feature

vectors, the training data set is used to learn the model and is applied on the test data to find the support. The results obtained from the CHIS organizers shown that the method adopted for identified for finding the relevance factor in our work was not producing effective when compared with other models submitted for this task. In the results obtained for task 2 our model was found to be working better and is effective to compute the support. It stood first when compared with the other models developed for this task.

6. REFERENCES

[1] Liu B. Sentiment analysis and opinion mining. Synthesis lectures on human language technologies. 2012 May 22;5(1):1-67.

[2] Pang B, Lee L. Opinion mining and sentiment analysis. Foundations and trends in information retrieval. 2008 Jan 1;2(1-2):1-35.

[3] Li Y, McLean D, Bandar ZA, O'shea JD, Crockett K. Sentence similarity based on semantic nets and corpus statistics. IEEE transactions on knowledge and data engineering. 2006 Aug;18(8):1138-50.

[4] Metzler D, Bernstein Y, Croft WB, Moffat A, Zobel J. Similarity measures for tracking information flow. InProceedings of the 14th ACM international conference on Information and knowledge management 2005 Oct 31 (pp. 517-524). ACM.

[5] Grossman DA, Frieder O. Information retrieval: Algorithms and heuristics. Springer Science & Business Media; 2012 Nov 12.

[6] Huang A. Similarity measures for text document clustering. InProceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand 2008 Apr 14 (pp. 49-56).

[7] Meyer D, Wien FT. Support vector machines. The Interface to libsvm in package e1071. 2015 Aug 5.