Multilingual Information Filtering by Human Plausible Reasoning

Asma Damankesh[1], Farhad Oroumchian[1], and Khaled Shaalan[2]

[1] University of Wollongong in Dubai, P.O. Box 20183, Dubai, UAE
[2] The British University in Dubai , P.O. Box 502216, Dubai, UAE
adamankesh@acm.org,
FarhadOroumchian@uowdubai.ac.ae,
khaled.shaalan@buid.ac.ae

Abstract. The theory of Human Plausible Reasoning (HPR) is an attempt by Collins and Michalski to explain how people answer questions when they are uncertain. The theory consists of a set of patterns and a set of inferences which could be applied on those patterns. This paper, investigates the application of HPR theory to the domain of cross language filtering. Our approach combines Natural Language Processing with HPR. The documents and topics are partially represented by automatically extracted concepts, logical terms and logical statements in a language neutral knowledge base. Reasoning provides the evidence of relevance. We have conducted hundreds of experiments especially with the depth of the reasoning, evidence combination and topic selection methods. The results show that HPR contributes to the overall performance by introducing new terms for topics. Also the number of inference paths from a document to a topic is an indication of its relevance.

1 Introduction

Human Plausible Reasoning (HPR) is a relatively new theory that tries to explain how people can draw conclusions in an uncertain and incomplete situation by using indirect implications. For 15 years, Collins and his colleagues have been investigating the patterns used by human to reason under uncertainty and incomplete knowledge [1]. The theory assumes that a large part of human knowledge is represented in "dynamic hierarchies" that are always being modified, or expanded. This theory offers a set of frequently recurring inference patterns used by people and a set of transformations on those patterns [1]. A transformation is applied on an inference pattern based on a relationship (i.e. generalization and specialization) to relate available knowledge to the input query. Elements of expression in the core theory have been summarized in Fig. 1. The theory has many parameters for handling uncertainty but it does not explain how these parameters could be calculated which is left for implementations and adaptations. Interested readers are referred to references [1], [2] and [3]. Different experimental implementation of the theory such as adaptive filtering [4], XML retrieval [5] or expert finding [6] have proved the flexibility and usefulness of HPR in

C. Peters et al. (Eds.): CLEF 2009 Workshop, Part I, LNCS 6241, pp. 366–373, 2010.

the Information Retrieval (IR) domain. All the works on HPR shows that it is a promising theory which needs more investigation to be applicable. This research is about creating a framework for multilingual filtering and information retrieval where all aspects of retrieval in this environment are represented as different inferences based on HPR. In this framework, documents and topics are partially represented as a set of concepts, logical terms and logical statements. The relationships of concepts are stored in a knowledge base regardless of their language of origin. Therefore, by inference, we can retrieve relevant documents or topics of any language stored in the knowledge base. This paper is structured as follows. Sections 2 describes the system architecture. Section 3 explains the experimental configurations. Section 4 summarizes the results and section 5 is the conclusion.

Table 1. Elements of expression in The Core Plausible Reasoning Theory

Baghdad is the capital of Iraq	
Referent $r_1, r_2, ..$ or $r_1, r_2, ...$	e.g. Baghdad
Argument $a_1, a_2, ..$ or $F(a)$	e.g. Iraq
Descriptor $d_1, d_2, ..$	*e.g.* Capital
Term $d_1(a_1), d_1(a_2), d_2(a_3), ..$	e.g. Capital(Iraq)
Statement $d_1(a_1) = r_1, d_1(a_2) = r_1, r_2, .., d_2(a_3) = r_3, ..$	*e.g.* $Capital(Iraq) = Baghdad$
Dependency between terms : $d_1(a_1) \leftrightarrow d_2(a_2)$	
e.g. latitude(place) $\leftrightarrow$ average-temp(place): Moderate, Moderate,Certain	
(translation): i am certain that latitude constrains average temperature with moderate reliability and that the average temperature of the place constraints the latitude with moderate reliability	
Implication between statements : $d_1(a_1) = r_1 \leftrightarrow d_2(a_2) = r_2$	
e.g. grain(place)=rice,... $\leftrightarrow$ rainfall(place)=heavy: high, Low,Certain	
(translation): i am certain that if a place produce rice, it implies that the place has heavy rainfall with high reliability, but if a place has heavy rainfall it only implies that the place produces rice with low reliability	

2 System Architecture

System architecture is depicted in Fig. 1. The Text Processor unit processes a document into a set of concepts, logical terms and logical statements. Document Representation unit, assigns a weight to each term. Topic Retrieval unit finds topics that have been indexed by the given terms and computes a certainty value. Inference Engine applies transforms of Human Plausible Reasoning to the terms in the document representation that exist in the knowledge base and generates a set of new terms. These new terms are used to expand the document representation. Then the new terms are given to Topic Retrieval to retrieve matching

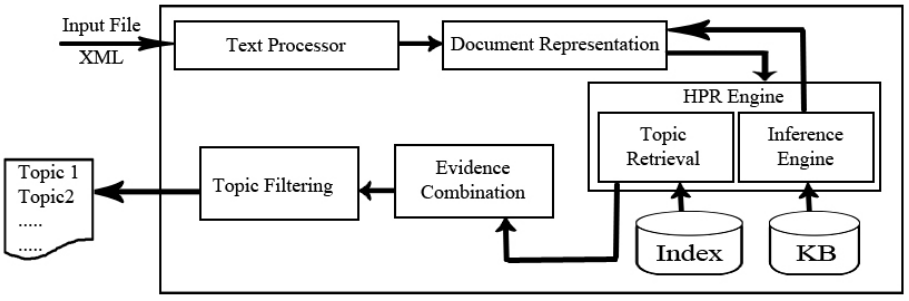

Fig. 1. Topic Retrieval and Filtering Unit

topics. The process of expanding the document representation and retrieving topics will be repeated several times. Each document term could be processed more than once (different inferences can reach the same term via different paths). A document could be linked to the same topic through multiple inferences and paths. Therefore, multiple certainty values could be assigned to a topic through different reasoning paths and terms. Topic Filtering unit is responsible for combining these certainty values into a single certainty value that represents the confidence in how well a topic can be inferred from a document representation. the KB is created by the Information Extractor Unit depicted in Fig. 2. This unit takes a list of file names and one by one reads through these files. Each file contains a document. The document goes through a pre-processing for normalizing the text. Then Part of Speech Tagging and stemming are applied. For POS tagging we have used Monty Tagger [7] and for stemming we have used a Python version of Porter stemmer [8]. The Text Miner is a rule based program that takes in the part of speech tagged text and extracts the relationships among the concepts. At the moment these rules are based on the clue words in the text. For example, they use propositions to infer relationships between two concepts around the proposition. To build the Knowledge Base, the Build KB unit takes in these relationships and calculates a confidence value based on the frequencies of occurrences of relationships.

The KB normally contains KO (Kind of), ISA (is a) and PO (Part of) relations. In case of cross language, we will have SIM (similarity) relationship which relates concepts with the same meaning from different languages together.

3 Experiments

The experiments were conducted on INFILE test collection. The collection consists of 100,000 documents out of which 1597 relevant pair of documents were provided for evaluation purposes. Because we did not have access to a separate text collection, we build our KB using the same INFILE test collection. This may or may not introduce a bias into the experiments but on the other

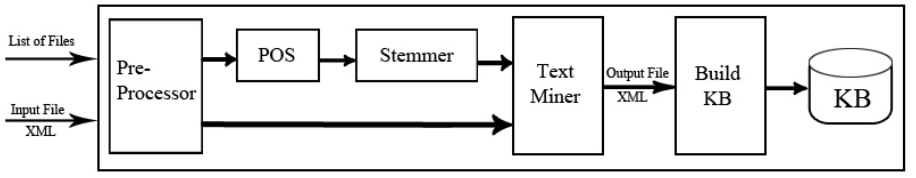

Fig. 2. Information Extractor Unit

hand what would be the benefit of testing our system with a KB with incompatible vocabulary. The KB only contains the relationships among the concepts and it does not contain any statistical information about the distributions of the concepts and their frequencies among the documents or topics. Therefore, the filtering hypothesis is not violated and the system is not able to make any assumption about the concepts and documents. In the rest of this section we describe different settings of the topic filtering process.

3.1 Concept Selection in Documents

Each document is treated as a query and is represented by a set of concepts $(Q_1, Q_2, ..., Q_k)$. These concepts are extracted from the heading and the content of the document. Only the concepts with a certainty more than the average threshold are used in document representation. Each concept Q_k is processed if $Freq_{Q_k} \geq 2$ and $\gamma_{Q_k} \geq avg$ where $avg = \sum \gamma_{Q_k}/N$. During the reasoning process new concepts will be generated and only those concepts that their certainty is bigger than the average certainty in the original document will be added to the representation and will be used in the later stages of reasoning for generating more concepts.

3.2 Evidence Combination

During the processes of reasoning multiple paths could be found between the concepts in a document and the concepts in a topic. Each path will have a certainty value which shows the system's confidence in that line of reasoning. These certainty values are combined in four steps to calculate the overall confidence in relevance of a document to a topic.

First level Combination: in each step of the reasoning if a concept is generated several times we keep only the one which has the maximum certainty value. In other words for $(inf_{i-t}, Q_{k-t}, topic_j, \gamma_j)$ take $max(\gamma)$.

Second Level Combination: many reasoning paths with the same type of inference or transform could relate a document concept to a topic concept. we only keep the path with maximum certainty value. In other words $(inf_{i-t}, Q_k, topic_j, \gamma_j)$ take $max(\gamma)$.

Third level Combination: for all the unique concepts in the document representation that have been related to the concepts in the topic through different

inferences $(inf_i, Q_k, topic_j, \gamma_j)$ calculate the sum of all the certainty values for all the paths connecting any document term to any topic term $(\sum \gamma - \prod \gamma)$ and return $(Q_j, topic_j, \gamma_j)$

Forth level Combination: for all unique concepts in the document representation that have been related to the concepts in topic query $(Q_j, topic_j, \gamma_j)$, calculate the sum of all the certainty values for all the paths connecting any document term to any topic term $(\sum \gamma - \prod \gamma)$ and return the list of retrieved documents $(topic_j, \gamma_j)$.

3.3 Topic Selection

The last component of our system is the Topic Filtering process. In this process, the system decides which one of the retrieved topics should be returned. We have conducted hundreds of experiments and have experimented with different factors that could influence this decision. One factor is the depth of the reasoning Process. During this process, the system traverses the concept hierarchies in the KB up and down to find new concepts that could be added to the document representation. Level indicates the number of levels that the system goes up and down the hierarchy using inference patterns. Level 0 means no inferences are applied on the concepts, i.e concepts have been matched against the topics directly. Another factor is M the number of topics we want to return for each document. $M = All$ means return all the topics that have been retrieved. We have experimented with $M = 1, 2, 3$ and All. Another factor is the confidence threshold. Some of the thresholds we have experimented with are:

No Threshold: no threshold means all certainty values are acceptable.

Threshold 1: $\gamma_{doc_j} \geq max(\gamma_{doc}) - \alpha * max(\gamma_{doc})$
In this case a topic is returned if its certainty value is within α percent of the maximum so far for that topic. The maximum is updated after each document.

Threshold 2: $\gamma_{doc_j} \geq avg(\gamma_{doc}) - \alpha * avg(\gamma_{doc})$
In this case, a topic is returned only if its certainty value is within α percent of the current average confidence value for that topic. Two different kinds of averages have been tried: A regular average so far and a monotonic average. The monotonic average is updating the average only by the increasing values of average.

Threshold 3: $\gamma_{doc_j} \geq min(\gamma_{doc}) - \alpha * min(\gamma_{doc})$
In this case a topic is returned only if its confidence value is within α percent of the minimum so far for that topic.

With 4 different M values and 3 levels and 5 different thresholds, we have conducted 141 different configurations. Each configuration has been tried with different values of α

4 Results

Table 2 shows the results for $Top1$ values of M with different levels and thresholds, with $\alpha = 0.7$ which gave the best results. In general, the min threshold

were the best threshold for certainty values. New documents were found at level 1 and 2 but not that many; so reasoning had a contribution when the level is either 1 or 2. It seems that using a certainty threshold is better than not using any thresholds. At level 1 for example, when $\alpha = 70$ is used, number of retrieved documents is dropped by 70 percent compared to when no threshold is used. This has increased the precision from 0.035 to 0.086. It seems that the similarity values of the relevant documents were below average but much higher than the minimum. This is an indication of a ranking problem.

Table 2. The results for $M = 1$ and $\alpha = 0.7$

Level	Threshold	Ret	Rel-Ret	Prec	Rec	$F-0.5$
2	$None$	19741	467	0.0347	0.2144	0.0532
	$\gamma_d \geq [max(\gamma) - 0.7 * max(\gamma)]$	11669	108	0.0405	0.1376	0.0553
	$\gamma_d \geq [\frac{\sum \gamma}{N}]$	6378	211	0.0562	0.0965	0.0571
	$\gamma_d \geq [min(\gamma) - 0.7 * min(\gamma)]$	6036	251	0.0742	0.1081	0.068
1	$None$	19616	461	0.0351	0.2114	0.0534
	$\gamma_d \geq [max(\gamma) - 0.7 * max(\gamma)]$	12326	313	0.0387	0.1377	0.0538
	$\gamma_d \geq [\frac{\sum \gamma}{N}]$	6469	230	0.0639	0.1029	0.063
	$\gamma_d \geq [min(\gamma) - 0.7 * min(\gamma)]$	5931	239	0.086	0.1073	**0.073**
0	$None$	17074	463	0.0409	0.2148	0.06
	$\gamma_d \geq [max(\gamma) - 0.7 * max(\gamma)$	14528	421	0.043	0.199	0.061
	$\gamma_d \geq [\frac{\sum \gamma}{N}]$	6170	242	0.0608	0.1316	0.0661
	$\gamma_d \geq [min(\gamma) - 0.7 * min(\gamma)]$	3937	198	0.0922	0.0923	0.0676

Table 3 shows the relationship among a number of inferences that has been used and a number of relevant documents retrieved. The number of inferences is an indirect indication of the length of the inference path. One inference means depth of $level0$ and direct match. However, since we have only levels $0, 1$ and 2 that means in each of these levels topics were retrieved through multiple inference paths. Basically, the more inference we go through the higher precision is we get but most of relevant retrieved topics were found in direct matching.
Table 4 shows how the number of matching terms between document and topic, are related to their precision. From Table 4, we observe that the more terms matches the higher the precision.

Based on the above observations we have run more experiments by combining the number of inference paths, the number of terms and certainty into a retrieval criteria. A few of these runs have been depicted in Table 5. Combining the number of terms matched with the number of inference paths or number of terms and certainty threshold has resulted in better performance and decreasing the number of retrieved documents. Although *precision* and *recall* and $F-measure$ are very important evaluation factors but showing as few documents to user

Table 3. Number of Inference paths and precision

Num of Inference paths	Ret	Rel-Ret	Prec
1	36801	290	0.007
2	7833	123	0.015
3	1944	100	0.051
4	584	34	0.058
5	138	16	0.115
6	55	11	0.2
7	20	4	0.2
8	11	4	0.36
9	5	1	0.2
10	1	0	0.0

Table 4. Number of terms matched between documents and topics and precision

Num of Terms	Ret	Rel-Ret	Prec
1	44400	343	0.007
2	2468	153	0.061
3	425	60	0.141
4	65	18	0.276
5	15	9	0.6
6	1	1	1

as possible also is important and improves user satisfaction in real systems. However, from these experiments it seems that there is an information overlap between the number of terms and the number of inferences which limits the usefulness of the approach.

Table 5. Number of Inference paths and precision

Num of matching terms	Num of inference paths	Certainty threshold	Rel-Ret	Ret	Prec	Rec	$F-0.5$
> 1	> 2		244	4054	0.061	0.153	**0.088**
> 1		> 0.3	304	5485	0.056	0.191	**0.087**
< 1 or	> 1		294	10591	0.028	0.185	0.049

A major problem we noticed in expansion of concepts is the lack of sufficient relations in KB. Also we felt the concepts weights generated from text processing, are not good representative of their value for the documents. Our conclusion from all these experiments was that, we need to investigate more text processing aspects and create better knowledge base. Once a better KB is built, we can work on certainty calculations and evidence combination. Also more sophisticated thresholds need to be experimented for both term and topic filtering.

5 Conclusion

We have built a system which uses inferences of the theory of Human Plausible Reasoning to infer new terms for expanding document representation and the relevance of a document to a topic in a filtering environment. We represent all the concepts regardless of their language of origin in the same knowledge base. Therefore the same inferences can retrieve any topic from any language

in response to arrival of a document. We used English INFILE text collection to build our KB and then we conducted hundreds of different experiments with different configurations. The recall of the system is less than what we expected and we can trace this back to the text processing unit and specially to the Text Miner unit. In future, we need to work on three aspects of our system, text processing, certainty calculations and evidence combination. Specially, we need to improve the quality of relations, we extract as they have a direct effect on recall of the system.

Acknowledgments. We would like to thank University of Wollongong in Dubai specially Information Technology and Telecommunication Services (ITTS) department as well as Dr Sherief Abdellah for providing us with a server to run our experiments.

References

1. Collins, A., Michalsky, R.: The Logic Of Plausible Reasoning A Core Theory. Cognitive Science 13, 1–49 (1989)
2. Collins, A., Burstein, M.: Modeling A Theory Of Human Plausible Reasoning. Artificial Intelligence III (2002)
3. Darrudi, E., Rahgozar, M., Oroumchian, F.: Human Plausible Reasoning for Question Answering Systems. In: International Conference on Advances in Intelligent Systems - Theory and Applications AISTA 2004 (2004)
4. Oroumchian, F., Arabi, B., Ashouri, E.: Using Plausible Inferences and Dempster-Shafer Theory of Evidence for Adaptive Information Filtering. In: 4th International Conference on Recent Advances in Soft Computing, pp. 248–254 (2002)
5. Karimzadehgan, K., Habibi, J., Oroumchian, F.: Logic-Based XML Information Retrival for Determining the Best Element to Retrive. In: Fuhr, N., Lalmas, M., Malik, S., Szlávik, Z. (eds.) INEX 2004. LNCS, vol. 3493, pp. 88–99. Springer, Heidelberg (2005)
6. Karimzadehgan, M., Belford, G., Oroumchian, F.: Expert Finding by Means of Plausible Inferences. In: International Conference on Information and Knowledge Engineering, IKE 2008, pp. 23–29 (2008)
7. Hugo, L.: MontyLingua: An end-to-end natural language processor with common sense (2004), `http://web.media.mit.edu/~hugo/montylingua`
8. Porter, M.F.: An algorithm for suffix stripping. Program 14(3), 130–137 (1980)