Intelligent Search Tool for Business Analytics

Harish V. Jadhao
Department of Computer & Information Technology
College of Engineering Pune, India.
Email: jadhaohv10.comp@coep.ac.in

Dr. Jagannath Aghav
Department of Computer & Information Technology
College of Engineering Pune, India.
Email: jva.comp@coep.ac.in

Prashant Gaikwad
John Deere Technology Center Indian(JDTCI), Pune, India.
Email: GaikwadPrashant@johndeere.com

***Abstract*—Business Intelligence (BI) requires integrated environment for knowledge acquisition and aggregation that enables the collection analysis and interpretation of data to reveal anomalies, key entities and relationships. The massive amount of information available for business analysis brings new challenges, and so far classical BI does not consider the meaning of data that limits data analysis.**

In this Paper, we propose solution for Business Analytics (BA) using Semantic Technology, which aims at finding, gathering, aggregating and analyzing information to identify and analyze the market need, market segmentation and competition. Semantic Technology is heavily relying on ontology that enables smooth data integration, rigorous knowledge representation, efficient querying and hypothesis generation. This paper details the steps to transform textual resources into domain concept maps, and offers integrated, robust and flexible presentation layer to Business Analyst. This helps Business Analyst to visualize domain concept using dynamic and interactive network graph that allows visually explore ideas and information.

***Index Terms*—Ontology based information Extraction, Linguistic Analysis, Semantic Analysis, Dependency structure**

I. Introduction

Business Intelligence is the process of collection, acquisition, aggregation, interpretation and assessment of information for decision making. It is the way of arriving at optimal or realistic decision by applying "branch of logic" with analysis to existing data.

Business Analyst is an expert who gathers the data about marketing field from legitimate sources in order to provide valuable information used in decision making. After a thorough research by adapting both quantitative and qualitative research techniques, the collected data is prepared for analysis and interpretation. Market research provides important information to identify and analyze the market needs, market value, market segmentation and competition. Usually, to make beneficial decisions from the business point of view, Business Analyst has to go manually through each document. It requires large amount of manual efforts. These manual efforts are time-consuming and energy intensive. For example, at present the organization's web site contains important legacy market data describing Organization, Country, Industry and Product. This data is distributed and provided in heterogeneous formats (e.g. HTML, XML, PDF, RSS feeds etc.), making it hard for Business Analyst to make sense out of the data.

In this Paper, solution for Business Analytics has been proposed that aims at processing dynamic data and provide appropriate, compact size information for Business Analyst. Business Analyst uses this information to identify and analyze the market needs, market value and competition, thus reduces manual efforts. Business Intelligence takes advantage of newly emerging Semantic Technologies [4], advocated by semantic web based on semantic knowledge base and NLP technologies. These technologies enhance data access, information analysis and allow Business Analyst to find relevant content more efficiently that improves decision making. In order to solve Business Analyst's problem, it is necessary to convert legacy data from its current form into a machine-readable representation that linked into the Web of Linked Data. This is done by linguistic analysis, semantic analysis and explanatory graphics that gives you contextual information in marketing field. This process recognizes entities like companies, industries, products, regions and phrases, and understands how they are related to each other and puts them into context.

Here, we concentrate on the application of ontology-based annotations over the given content with regards to the static domain ontology in the context of Business Intelligence. Ontology based information extraction is the process of identifying entities, properties and relations based on domain static ontology. GATE [23] platform provides a set of processing resources for development of ontology based information extraction system. Ontology contains concepts arranged in class/sub-class hierarchies (e.g., John Deere is a type of Company), relations between concepts (e.g., John Deere announce massive tax hits due to ObamaCare), and properties (e.g., John Deere is leading manufacturer of construction equipment and forestry equipment). We are developing static ontology with the help of domain expert which represents application of domain and capture the expert's knowledge. We are working with the static ontology is being developed for an agriculture and technology application in the agriculture marketing domain where the objective is to model information about Company, Country/Region, Industry, and Products. Domain experts are identified 'John Deere', 'Caterpillar' as Company, and 'Material handler', 'Gear' as Product, etc.

A very first step of our proposed architecture is to convert unstructured data into structured data. Structured data represents knowledge as a set of entities within a market domain and the relationships between those entities. To develop an intelligent search tool for Business analytics, it is very important to build systematic information extraction framework for

ontology learning. Ontology learning [2] is semi automatically extract relevant entities and relations between them from a unstructured text data, and represent this abstract structure into a RDF (Resource Description Framework) graphs. These can be further stored in RDF knowledge base and queried using SPARQL query language. During the ontology learning process, extracted structured information (i.e. entities, terms and relations) from the legacy data based on static ontologies are semantically represented in RDF graphs. Business analyst makes decision quickly and confidently by analysing structured data. This can be done by providing the specific query on corresponding RDF of unstructured text to find, manage and analyze information. The results of query are visualized using network graphs that help BA to go through relevant information of interest.

In Network graph, node represents entity, term, phrase or concepts and link between two nodes represents the relations between the respective nodes. By lingering over the any relationship Business Analyst can easily see the articles that explain the association.

II. Related Work

Ontology [1] learning and population from unstructured data are two very active research fields. Obviously most of the work had been carried out in same direction within Artificial Intelligence, Machine Learning and NLP. Ontology learning primarily focused on defining the concepts and relations between them. Thus, it aims at extracting domain terms, concepts, individuals, concept attributes and relations from textual data.

Ontology Learning from text relies on different approaches such as: 1) Machine Learning and statistical method to learn rules from annotated corpus. 2) Linguistic methods to discover information extraction rules by inspection of a corpus. 3) Combination of both i.e. linguistic and machine learning methods. The key advantage of linguistic method over statistical method is it does not require large amounts of training corpus. Which is often expensive to acquire. On the other side 1) rule identification process is tedious and laborious 2) domain adaptation may require significant reconfiguration.

In statistical approach, domain adaptation is relatively ease due to automatic rule induction. It shows decent results for entity annotation, such as identifying gene name in system biology [5]. But this approach is not effective in case of relation identification due to lack of annotation text corpus. Moreover training data may be hard to acquire. Significant change in domain specification requires re-annotation of entire training data. This needs repeating the training process for each new domain in order to be accurate.

Maedche, Staab and Volz [6] present a dynamic approach for construction, reuse and maintenance of ontology from domain text by applying statistical approach(Text-to-Onto tool). This work addresses combination of two approaches: 1) linguistic techniques 2) statistical techniques to simplify and accelerate ontology construction. The overall system involves filtering and preprocessing textual data. This is carried out by processing resources component and NLP system, then modelling properties are identified by applying machine learning algorithm. This application also provide ontology refinement, pruning. It focuses on learning the meaning of unknown words over the time. Related projects include: Text-2-Onto [7], OntoLearn [8], OntoLT [9], OntoGen [10] tools developed to support the user in constructing ontologies from a textual data.

As a result, there has been considerable focus on rule based approach, even though requiring a lot of manual work, proves to be more effective and transparent in capturing the semantic criteria.

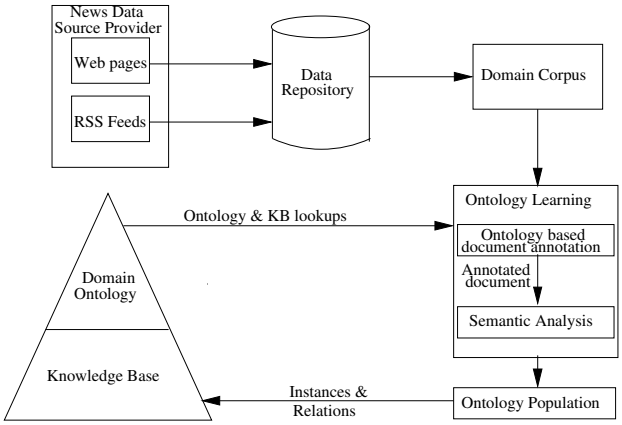


Fig. 1. System Architecture

III. Proposed Method

In this paper, we propose a novel framework for BI based on semantic and NLP technologies. Information Extraction (IE) is a key NLP technology for automatically identifying and extracting specific types of information from text (e.g. different types of Entities and relation between them). Without an IE system, business analysts would have to read hundreds of web documents, news articles and tabular data to manually extract the necessary information for identify and analyze the market.

Fig. 1 shows the architecture diagram, which is composed of static ontology and rule based approach for information extraction from the unstructured text. In this approach, information extraction is the process of identifying relevant entities, properties, and relations between them from unstructured text based on domain ontology. We are having static domain ontologies represents Knowledge Base(KB). This represent the domain of application and capture the expert's knowledge. Business Analyst gathers data from legitimate sources and stored in Data Repository. News data is an information source for Business Analyst. Data Repository is a database that holds the RSS feeds, Web sites URLs and news articles. These are RSS feeds or web site's URLs used by Business Analyst to short list the information of interest.

A. Domain Corpus

Data Repository contains complete news articles. These news articles are validated against the static domain and further carry out the structural analysis. Structuring analysis is

performed to remove noisy elements, such as scripting code and HTML comments, while retaining metadata and enriching information. This information can be headings, subheadings, lists, paragraphs, etc. The output of this module is complete text news articles. These articles consist of metadata in XML format required for further processing.

B. Ontology Learning

Ontology learning extract relevant entities and their relations from unstructured text corpus, and transform this abstract structure into a formal domain RDF graphs. In order to extract the entities and their relations from the document, we are carrying out linguistic analysis followed by semantic analysis. These analysis are performed using domain ontology, that has been developed through interaction with domain experts. Domain ontology consists of set of concepts and known relationships within domain of interest.

1) Linguistic Processing: We are carrying out linguistic processing using GATE [23] tool architecture. It is a publicly available tool that allows users to define rules for creating different annotation over the text data.

Linguistic analysis is performed on news articles by processing an Information Extraction Pipeline. This pipeline is create with GATE architecture, consists number of processing resources executes on text data in predefine order. It has two phases 1) Linguistic analysis and 2) Semantic Analysis. Linguistic analysis has six processing resources: 1) Tokenizer, 2) Sentence Splitting, 3) Part of Speech Tagger, 4) Morphological analyzer, 5) Onto Root Gazetteer, 6) JAPE Transducer. Onto Root Gazetteer takes domain ontology as input and produces annotated corpus with the domain entities. Further, JAPE Transducer executes specified rules over the annotated corpus based on regular expression. For example rules for identifying different attributes like money, time, etc. We defined some set of JAPE rule for capturing relevant entities required for anaysis. Each article from domain corpus is processed by IE Pipeline using static ontology. During the processing annotations are created on the document. The output of linguistic processing is annotated corpus with people, product, places, organization, companies and static relations specified in domain ontology. Further output of linguistic analysis is subjected to semantic analysis. Semantic analysis captures an unknown relation that appears in the text data between annotated entities.

2) Semantic Processing: Semantic analysis of the text is the key issue for relations extraction form unstructured text. This analysis is deeply intertwined process of syntactic and semantic analysis. Semantic analysis processes the text sentence by sentence, generating the parse tree and dependency structure for each sentence. We are carrying out semantic processing using Stanford Parser [17] tool architecture. It allows defining simple set of rules (pattern rules) for relation extraction using dependency relations that will covered in the next section.

The main component of semantic analysis are parsing, dependency generation and relation extraction. Semantic analysis generates parse tree and dependencies for each key sentence coming after linguistic processing. In order to identify key sentences, we simply define JAPE rules that run over the annotated corpus. For example JAPE rule for key sentences as "sentences containing two different Entities and Verb Phrases in between them". Such sentences act as key sentences and further subjected to semantic analysis. The output of semantic processing is collection of triplets. These generated triplets validate against domain description and assign a semantic role to them by using domain ontology.

Structure Analysis: Despite the lack of semantics in unstructured text, for any task involving information extraction from text, employs linguistic parsers [17] to perform syntactic analysis of text. Parsing process results in the generation of parse trees. Parse tree is an ordered and rooted tree that represents the syntactic structure of a sentence.

Dependency Generation: Dependency parse trees as a means for semantic relation extraction among entities. It is a directed graph reveals implicit dependencies between objects (words) within sentence i.e. between words that are far apart in a sentence. It provides a useful structure for sentence by annotating edges with dependency types e.g. modifiers, auxiliaries, modals, verbs and adverbs. Simplified definitions of the set of dependencies (grammatical relationships) are described in the Stanford typed dependencies manual [17]. The current representation contains approximately 53 grammatical relations. These dependencies are all binary relations: grammatical relation holds between a governor (also known as a regent or a head) and a dependent. Here we list few relations that are important in our context.

nsubj: nominal subject
A nominal subject is a noun phrase which is the syntactic subject of a clause. The governor of this relation might not always be a verb: when the verb is a copular verb, the root of the clause is the complement of the copular verb, which can be an adjective or noun.

dobj: direct object
The direct object of a VP is the noun phrase which is the (accusative) object of the verb.

ccomp: clausal complement
A clausal complement of a verb or adjective is a dependent clause with an internal subject which functions like an object of the verb, or adjective.

amod: adjectival modifier
An adjectival modifier of an NP is any adjectival phrase that serves to modify the meaning of the NP.

acomp: adjectival complement
An adjectival complement of a verb is an adjectival phrase which functions as the complement (like an object of the verb).

nn: noun compound modifier
A noun compound modifier of an NP is any noun that serves

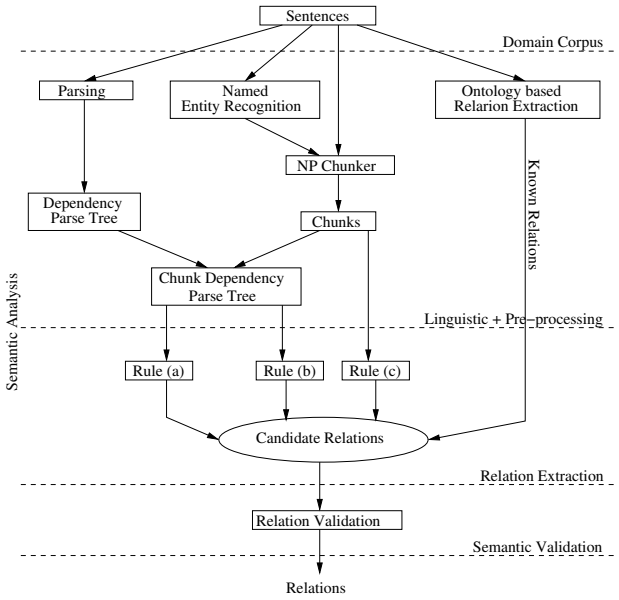


Fig. 2. Relation Extraction Framework

to modify the head noun.

Relation Extraction: Relation extraction process makes use of small set of simple rules. Rule is a simple assertion containing a set of premises and consequents. By definition, a rule warrants the execution of actions defined in the consequents whenever the conditions defined in the premises hold. These set of rules are building upon open source tools applied for noun phrase chunking and dependency parse tree. Fig. 2 shows Relation Extraction Framework, which uses dependency parse tree as means for generic relation extraction. Here, we extract candidate relations by extracting paths such that it connects pairs of entities from dependency parse trees. These paths contain the relevant terms describing the relation between the given pair of entities.

Normally sentence of news text is long and complicated and frequently mentions a number of possible named entities. These named entities are identified during linguistic processing. We are not looking for any particular kind of interaction among the possible entities. Currently, we described following set of possible rules applied on dependency parse tree, chunk dependency parse tree and noun phrase sentence that extracts candidate relations and further subjects to semantic validation step.

Rule 'a': nsubj - verb - dobj/pobj
Rule 'b': nsubj - verb - [(nsubj - verb - dobj/pobj) / (nsubj - ccomp - xcomp)]
Rule 'c': NounPhrase(NP) - VerbPhrase(VP)- NP.

Rule 'a' extracts the relations among name entities in the dependency parse tree of simple sentence. A simple sentence expected to have one independent clause and no dependent clauses. Independent clauses contains one subject and one predicate. It might contain modifier such as adjectival modifier, noun compound modifier, or quantifier phrase modifier. Because, modifier does not change the grammaticality of the sentence. But, it is not expected to have inner clauses, multiple subject or object.

Predi = {Relation | Node with two outgoing edges with labels "nsubj" and "dobj"}

NEntitySubject = {Entity | Node containing named entity; which is connected to the predi by edge with label "nsubj"}

NEntityObject = {Entity | Node containing named entity; which is connected to the predi by edge with label "dobj"}

Sentence describing *NEntitySubject* and *NEntityObject* are noun phrase chuck containing named entities (e.g. Person, Region, Country, etc.). *Rule 'a'* is simply extracts paths in the chunk dependency tree that lead from a start-point (generally the *NEntitySubject*) to an end-point (generally the *NEntityObject*). In Fig. 3, *Rule 'a'* extract the path as an information constructs if labels nsubj, dobj occur along a path in dependency graph connecting *Predi*, *NEntitySubject* and *NEntityObject*.

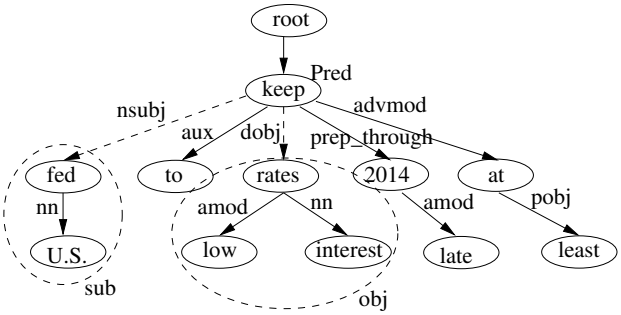
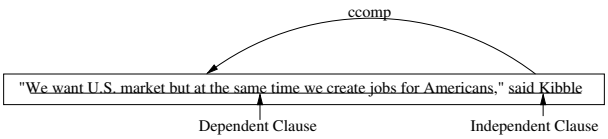


Fig. 3. Dependency parse tree is derived from stanford parser, showing words(eclipse), depedencies(edge pointing from head node to dependent node), dependency type(label on edge) and head of the sentence(root). Dotted lines indicate paths that are extracted by *Rule 'a'*.

Rule 'a' applied on the sentence '*U.S. Fed* to *keep low interest rates* at least through late 2014.' extracts the parts marked in italics as candidate relation shown in Fig. 3.

Rule 'b' extracts constructs from the complex sentence. A complex sentence is an assertion made by some physical entity that expected to have independent and dependent clause. For ease of understanding, we illustrate this case with descriptive example as follows. Consider a statement "We want U.S. market but at the same time we create jobs for Americans," said Kibble'. This statement is taken from news article, makes an assertion about 'US market'. Information extraction system operating in marketing domain recognized this statement as key information constructs about entities like U.S. market, America and Kibble. It has main subject that refer to dependent clause through verb. Such a relationship is nicely captured by casual complement 'ccomp' relation in terms of Stanford dependency relations.

In such type of sentences, dependent clause is often interpreted as object of main verb of the statement. Dependent



clause is itself as a sentence on which we applied *Rule 'a'* and extract information constructs. In such cases, dependent clause is a triplet (*NEntitySubject-Predi-NEntityObject*) and a object of the main subject of sentence. Finally such complex relations are subjected to RDF representation.

After processing each key sentence by NP chunker, *Rule 'c'* is applied on it and extract candidate relations. In this case, sentence containing noun phrases which has been recognized during NP chunking act as a subject and object. Thus by applying *Rule 'c'* on sentence extracts candidate relation connecting these two noun phrase chunks.

Form the above mentioned rule set, *Rule 'a'* and *Rule 'b'* are applied on chuck dependency parse tree, and *Rule 'c'* directly applied on sentence after NP chunking. These set of rules had been executing on key sentences in predefine priority for extracting the relations among entities. Finally, all extracted entities and relations between them subjected to semantic validation and RDF representation.

3) Information Validation: All generated constructs in the previous task are validated against the domain ontology and assign semantic role to them. We formulate a methodology to represent these validated information constructs using existing Resource Description Framework (RDF) specification. RDF specification is widely accepted and consumable. This is meta-data containing entities and relations between them. Those presents in the collection of triplets. All RDF data generated from unstructured document stored in Knowledge Base i.e. in triplet store (e.g. Sesame, Jena triplet store). Motivation behind the RDF representation is its enable the possibility of complex querying on the extracted information. We can easily get answers to complex queries. For example, which entity appears in most facts? Or which entities are involved in relation A?

C. Information Visualization

All rich information represented in RDF standard is visualized using network graph to business analyst. All information present in the RDF that enables BA to make complex queries on it. These queries are forming with SPARQL query language. BA can analyse the data with entity based and SPARQ query based search. SPARQL queries can be fired on a single Jena model, loaded with generated RDF and static ontology. In Entity based search, BA ask for particular entity, and gets set of entities and relations are related to search entity as a response. All resulted information are visualize through network graph. In network graph, node represents entity, term, phrase or concept, and link between two nodes represents the relations. Network graph visualizes relationships or associations based on analyst's search. By lingering over any relationship user can easily see the article that explain association.

IV. Conclusions

In this paper we have proposed a framework for Business Intelligence based on semantic technologies, advocated by semantic web. It is a system that extracts knowledge automatically, populates ontology with knowledge triplet and reassembles into the knowledge map. Knowledge map is knowledge representation that reveals the underlying relationships among different entities and visualize them using interactive and dynamic graph.

This system uses different open source tools. It is straight forward to implement and achieve competitive performance. It is mostly rely on available GATE tool's plug-in and Stanford parser. Future work on Intelligence search tool will continue to build domain ontology to refine its information extraction and consolidation process.

References

[1] Heiner Stuckenschmidt, Frank van Harmelen. *Information Sharing on the Semantic Web*, Springer, 2005.
[2] Paul Buitelaar, Philipp Cimiano. *Bernardo Magnini Ontology Learning from Text: An Overview*, IOS Press, 2003.
[3] S. Huffman. *Learning information extraction patterns from examples*, Workshop on new approaches to learning for natural language processing, IJCAI-95: 127142, 1995.
[4] D. Allemang, J. Handler. *Semantic Web for the Working Ontologist*,Elsevier, 2011
[5] Erick Antezana, Ward Blonde, Aravind Venkatesan, Bernard De Baets, Vladimir Mironov ,Martin Kuiper. *Semantic Systems Biology: enabling integrative biology via Semantic Web technologies*, ACM, 2011.
[6] A. Maedche, E. Maedche, R. Volz. *The ontology extraction maintenance framework text-to-onto*, In Proceedings of the ICDM01 Workshop on Integrating Data Mining and Knowledge Management, 2001.
[7] P. Cimiano, J. Volker, *Text2onto - a framework for ontology learning and data-driven change discovery*, 2005.
[8] P. Velardi, R. Navigli, A. Cucchiarelli, F. Neri. *Evaluation of OntoLearn, a methodology for automatic population of domain ontologies*, IOS Press, 2006.
[9] P. Buitelaar, D. Olejnik, M. Sintek. *A protege plug-in for ontology extraction from text based on linguistic analysis*, In Proceedings of the 1st European Semantic Web Symposium(ESWS), 2004.
[10] B. Fortuna, M. Grobelnik, D. Mladenic. *Background Knowledge for Ontology Construction*, Proc. 15th Int'l Conf. World Wide Web(WWW '06), pp. 949-950, 2006.
[11] Amal Zouaq, Roger Nkambou. *Evaluating the Generation of Domain Ontologies in the Knowledge Puzzle Project*, IEEE, 2008.
[12] Amal Zouaq, Roger Nkambou *present Building Domain Ontologies from Text for Educational Purposes*, IEEE, 2004.
[13] Amal Zouag, Michel Gagnon, Benoit Ozell. *Semantic Analysis using Dependency-based Grammars and Upper-Level Ontologies*, IJCLA VOL. 1, NO. 1-2, PP. 85-101, 2010.
[14] Adam Pease, Ian Niles, John Li. *The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications*, In Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web, Canada (2002).
[15] *Part-of-speech tagging*, http://en.wikipedia.org/wiki/Part-of-speech tagging.
[16] G. A. Miller. *WordNet: A Lexical Database for English*, Communications of the ACM, Vol. 38, No. 11: 39-41, 1995.
[17] *The Stanford Parser: A statistical parser*, http://nlp.stanford.edu/software/lex-parser.shtml.
[18] *An Introduction to Jena RDF API*, http://jena.sourceforge.net/tutorial/RDF API/index.html.
[19] *Resource Description Framework (RDF)*, http://www.w3.org/RDF.
[20] *SPARQL*, http://www.w3.org/TR/rdf-sparql-query.
[21] *Web Ontology Language (OWL)*, http://www.w3.org/TR/owl-features.

[22] *Domain and upper ontologies*, http://en.wikipedia.org/wiki/Ontology (information science).
[23] *GATE, General Architecture for Text Engineering*, http://gate.ac.uk/conferences/training-modules.html.
[24] *Kea, Automatic Keyphrase Extraction*, http://www.nzdl.org/Kea/index old.html