Context dependent keyword extraction for Image Retrieval based on Descriptive Metadata

M.Tech Project Report

Phase I

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology

by

Jayaprakash S

Roll No: 123050045

under the guidance of

Prof. Dr. Pushpak Battacharya

Department of Computer Science and Engineering

Indian Institute of Technology, Bombay

Mumbai

Acknowledgement

I would like to thank my guide, Prof. Dr. Pushpak Battacharya for the consistent support and guidance he provided throughout the semester. Further I would thank Mr. Krishanu Seal, Senior Director and Mrs. Prateeksha Chandraghatgi, Principal Technologist from Yahoo!, India for giving oppurtunity work with them and their constant support.

Finaly I would like to thank sandhan search team Mr. Arjun and Mr. Swapnil for valuable guidance and discussions.

Abstract

The aim of this project to retrieve an image which is relevant for a given news article which does not have annotated image. Reason for finding an image for a text, image visual tend to give the instant sense of contentment to user compared to text document. Further, as we see good fraction of news artciles does not image annotated with them. *Our approach* is to extract relevant keywords from news article given as text document, and using the set of keywords to finding an image for that article. We are experimenting with various supervised and unsupervised approaches.

CHANGE - ADD IF NEEDED

List of Figures

Contents

Chapter 1

Introduction

Use a picture. It’s worth a thousand words.

Arthur Brisbane

In this project, we are trying to generate the keywords for a given news document that is closely match with descriptive metadata of an relevant image. Intent of doing keywords extraction is to retrieve an relavant image for a given news article without annotated image. Once keyword is extracted from a text, keywords will be used for retrieving image using Image Search System or Engine.

1.1 Motivation

We live in Information Age, also known as the New Media Age. In this era, Information is freely exchanged and knowledge is easily accessed and, it is represented in several formats like text, images, videos, etc. More importantly, each of these representation convey information to the users at different rate and An image likely to provide instance sense of contentment than text.

People are genetically wired to respond differently to visuals than text. For example, humans have an innate fondness for images of wide, open landscapes, which evoke an instant sense contentment.

The motivation behind finding an relevant image for a news article are,

- **Increased user satisfaction**

  If a image is attached with news text, by looking at the image user can guess about the story in that news item. Hence avoids reading text to make decision on whether to read it or not. Further a better image may induce user to read the news article.

- **Content colloboration**

  Linking different representation of content coveys same infromation.

why keyword extraction?

1.2 Applications of Keyword extraction

Keyword extraction is widely considered as a subtask of information extraction. The motivation of keyword extraction is used to automatically extract important terms from a text document. Keyword extraction has many application apart from image retrieval, such as

- Recommendations
- Summarization
- Ad-relevance
- Document-similarity

1.3 Challenges

- News document is large in size whereas metadata is small. Searching by entire document text may not be viable and it may be not fetch an image.
- Scalability of content-based image retrieval systems.
- If we go ahead with keyword extraction technique for fecthing an image, deciding top-K is very difficult for each document. Less number of keywords may fetch irrelevant images, and more keywords than required may not fetch any images.

1.4 Metadata

Metadata are defined as the data providing information about one or more aspects of the data, such as:

1. Purpose of data
2. Date and Time of creation
3. Author of data
4. Means of creation of data

A text document's metadata may contain information about how long the document is, who the author is, when the document was written, and a short summary of the document. A digital image may include metadata that describe how large the picture is, the color depth, the image resolution, when the image was created, and other data.

The term 'metadata' refers to 'data about data'. Widely the term 'metadata' is used for two different concepts.

'Structural metadata' is about the design and specification of data structures and more precisely it should be 'data about containers of data'.

'Descriptive metadata', on the other hand is all about the content of data i.e, instance of application data. In our problem, we are trying to utilize the descriptive metadata of images for retrieval.

1.5 Classification of Images

We tried to classify images annotated to the news articles, based on its relation to the content of the articles. Difficulty level of finding these Images automatically will vary based on its classification.

Representative Images

Image attached to an article is a representative for the content of documents. and descriptive metadata of image most likely to match the sentence or title from the text. Relatively easy to retrieve image if extracted keywords or phrases.

Example-Representative

News Text : Article is about oil spil

Annotated Image : Image with description with 'oil spil happened in '

Iconic Images

Sometimes though articles is not directly talking about the person/entity in the Image, because of it's popularity it is tagged with the article. It is difficult identify exactly same image.

Example-Iconic Images

News Text : Wasting Time Is New Divide in Digital Era.

Image Description : Facebook Picture.

News Text : Any article related to Bollywood news (remake).

Image Description : Salman Khan's Image.

Aspectual Images

Similar to iconic but image is related to event or aspect described in the article, but article need not necessarily mention the person or entity in image explicitly.

Example-Iconic Images

News Text : Article is about Kolkata Knight riders won trophy

Image Description : An image of Mr,Shahrukh Khan holding trophy.

News Text : for an article titled 'US threatens war while considering talks with Syria, Iran'.

Image Description : ObamaâĂŹs Pictures with Syria map.

Historical Images

Article speaks about event happened at certain time whereas images of past is tagged .

Example-Historical Images

News Text : An article is about bomb blast of zaveri bazaar.

Image Description : zaveri bazaar before bomb blast.

News Text : For an article about TempleâĂŹs renovation

Image Description : temple before renovation is being used in the article.

Partially Relevant Images

relevant to portion of the article but not to the whole article.

Example-Partially Relevant Images

News Text : Oil spillage near Hartlepool causes A19 chaos.

Image Description : Drivers faced long delays after a busy dual carriageway was closed for almost two hours due to a two-mile spillage of oil.

1.6 Overview of System

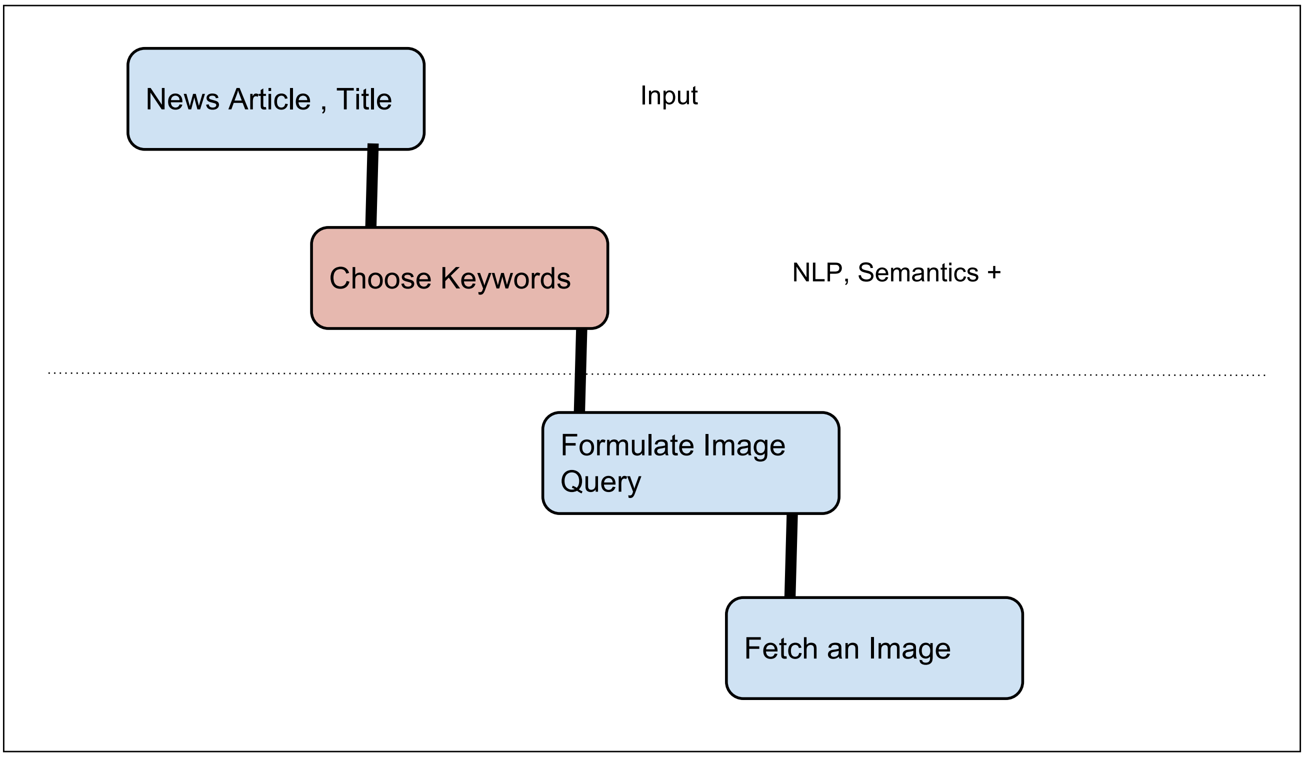

Figure 1.1: Overall System

Input: Input for the system consist of news article as text document with title.

Keyword Extraction: Input for the system consist of news article as text document with title.

Formulating Image Search Query: Image search query needs to formulated based on the keywords extracted.

Fetching an Image: Image Query formulated in the previous step must be used for firing search. An image from result set is used for annotating the news document. Optionally image metadata can be matched against the news document so that it could be

1.7 Contributions

TODO

1.8 Roadmap

The remaining part of this report organized as follows: Chapter 2 describes the literature study focusing on existing work on keyword extraction using supervised and unsupervised approaches. Chapter 3 is about the experimentations done including for keyword extraction. Chapter 4 is about Conclusion and Challenges of using Keyword extraction for image retrieval. Finally Chapter 5 is about the future plans.

Chapter 2

Literature Study

2.1 Unsupervised Keyword Extraction Methods

Unsupervised keyword extraction methods mostly rely on the relationship between the words in the text. Importance of the word is estimated based on exploiting the relationship to all other words.

2.1.1 TextRank

TextRank is a graph based unsupervised ranking algorithm formulated for text processing via extracting keywords and sentence extraction from documents. This method is based on Graph-based ranking algorithms like HITS algorithm [5] , Google Search Engine's PageRank [5] where is used for social networks, citation analysis and analysis of link structure world wide web. These graph based algorithms exploit the global information computed iteratively rather than looking at only surrounding information.

The basic idea of the graph based ranking models is 'voting' or 'recommendation'. If one vertice connected to another vertex, basically it is casting a vote for that other vertex. The higher number of votes the one vertex gets, the same amount of importance it gets. The importance of the vertex which is casting vote is going to determine the amount of importance should be given to that vote. Effectively the score associated with the vertex is determined by votes that are cast for it and scores of vertices that are casting vote for it.

Formally, considering G = (V, E) as directed graph with set of vertices V and set of edges E, where E is the subset of VxV. Let In(V_i) be the set of vertices that are voting V_i (predecessors) and Out(V_i) be the set of vertices that V_i is voting to. The score of vertex

V_i is calculated as follows [5],

$$S(V_i) = (1 - d) + d \sum_{j \in In V_i} \frac{1}{|Out(V_j)|} * S(V_j) \tag{2.1}$$

here d is called as damping factor that can be set between 0 and 1 initially, which has the role of adding a probability directly jumping from one vertex to another vertex (actually it signifies the default or implicit voting given by any vertex to all other vertices to avoid dead-end in random walk).

Starting from arbitrarily assigned values to all vertices in the graph, the vertex values are computed iterated until convergence or given threshold is reached. After convergence is reached, the scores of the vertices represents the importance of the vertices within the graph.

Document as graph

To use the graph ranking algorithm for natural language text, first we should build a graph that represent the given text and interconnected words and relations between them. Based on the application, text units of various sizes can be used as vertices (examples words, sentences, phrases etc). similarly based on the application we can decide type of relations should exist between vertices e.g. lexical or semantic relations, contextual overlap etc.

Overall steps of this algorithms is,

1. Identify the text units of document and add them vertices to the graph.

2. Figure out the relations between text units, that best suits the application. Edges that connects the vertices can be undirected or directed, Weighted or Unweighted.

3. Assign the initial scores of vertices arbitrarily and iterate through ranking algorithm until convergence.

4. Sort the vertices based on final score. and selects top-K vertices as candidate vertices or text units.

5. [Optional]Post processing is applied to vertices or textual units.

The expected result of keyword extraction task is set of keywords or phrases for a given natural language text. Any relation between two lexical units can be used as connection between two vertices. Here in this paper, co-occurrence relation with the controlled distance is used as edges or connection between vertices. Two vertices are said to be connected if that two lexical units tend to co-occur within a window of N words, where ‘N’ can vary.

The vertices added to graph contains lexical units of certain types, for instance in this paper [3] they have used the individual words as the vertices and connection between vertices (individual words) represents that they co-occurred in the text within the window size N.

Undirected Edges

Graph used for ranking keyword is undirected, whereas original algorithm was developed for directed graphs. If the two words tend to co-occur then they are mutually connected to each other, so each vetices in-degree equal to out-degree.

Weighted Edges

Edges in the TextRank model is weighted, they directly indicate the strength of connection between two vertices. In this if we two words tend to co-occur frequently then they will have strong (more weight) connection.

By considering above undirected and Weighted cases, the original graph based ranking algorithm has been modified into as follows,

$$WS(V_i) = (1 - d) + d * \sum_{j \in InV_i} \frac{W_{ji}}{\sum_{k \in OutV_j w_{jk}}} * WS(V_j) \tag{2.2}$$

Example-Text

Apple’s product road map is a topic that may receive more speculation than any subject in all of tech. With that in mind, some are expecting 2014 to be a very big year for the Cupertino, Calif.-based maker of the iPad and iPhone.

Jefferies analyst Peter Misek, he of the precarious Apple upgrade, says 2014 will indeed be a crucial year for Apple as the company lays out its next version of the iPhone, the iPhone 6.

Misek notes that the next phone will likely have a new design and a much bigger screen than its predecessor. A 4.8-inch screen is likely size. The iPhone 5s/5c has a 4-inch screen. ‘We discovered from Asian players that Apple is aggressively investing in OLED alongside its display partners,’ Misek wrote in his note. ‘Apparently Apple has begun to procure equipment for LG Display, Sharp, and Japan Display.’

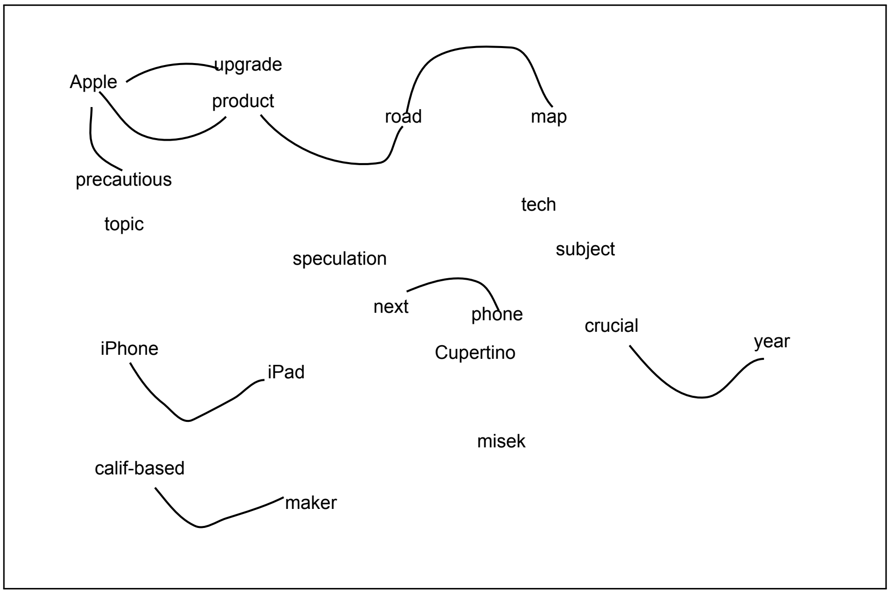

Figure 2.1: TextRank - Graph for the above text document

Process

First the given document was tokenized, and syntatic classes of each word (part of speech tag) is identified. It is said that picking only certain syntatic classes gives the better precision (nouns and adjectives). Only the unigrams considered as vertices. Graph ranking algorithm run on the constructed graph. Top fraction of vertices selected based on score given to vertices on convergence. If the selected unigram words tend to cooccur in the text they are combined together and considered as multi-word keywords or keyphrases.

Results

This algorithm was tested against 500 science articles where keywords algorithm was compared with manually annotated keywords. It is shown that this algorithm achieves highest F-score 36.2% when edges are considered as undirected with co-occurence window size (N) is 2.

2.1.2 RAKE

Rapid Automatic Automatic Keyword Extraction (**RAKE**) [6] is unsupervised, language independent method for extracting keywords from individual documents. RAKE is based on the assumption that keywords contains multiple words at large but very rarely contain stop words and punctuation in it.

RAKE needs stop words list, phrase delimiters and word delimiters as input parameters. Candidate keywords are chosen based on the stop words and phrase delimiters. Co-occurences of words within the candidate words used as measure for candidate keyword being a keyword.

First, the document is split into array of words based on the word delimiters. The resultant array is splited into sequence of continuous words based on the phrase delimiters and stop word occurence.

Example - Text Document

Criteria of compatibility of a system of linear Diophantine equations, strict inequations, and nonstrict inequations are considered. Upper bounds for components of a minimal set of solutions and algorithms of construction of minimal generating sets of solutions for all types of systems are given. These criteria and the corresponding algorithms for constructing a minimal supporting set of solutions can be used in solving all the considered types of systems and systems of mixed types.

Example - Candidate Keywords

Compatibility - systems - linear constraints - set - natural numbers - Criteria - compatibility - system - linear Diophantine equations - strict inequations - nonstrict inequations - Upper bounds - components - minimal set - solutions - algorithms - minimal generating sets - solutions - systems - criteria - corresponding algorithms - constructing - minimal supporting set - solving - systems - systems

TODO:EXAMPLE - scores

Example - Final Keyword Scores

minimal generating sets (8.7), linear diophantine equations (8.5), minimal supporting set (7.7), minimal set (4.7), linear constraints (4.5), natural numbers (4), strict inequations (4), nonstrict inequations (4), upper bounds (4), corresponding algorithms (3.5), set (2), algorithms (1.5), compatibility (1), systems (1), criteria (1), system (1), components (1),constructing (1), solving (1)

Scoring

After all candidate keywords are identified and graph of co-occurences is built. Score of a candidate keyword is calculated based on sum of it's member words scores.

Word scores are based on,

- word frequency *freq(w)*
- word degree *deg(w)*

- ratio of word frequency to degree *freq(w) / deg(w)*

deg(w) favours a word which occurs in longer candidate keywords. example . words that occur in many candidates are favoured by freq(w). example . Words that largely part of longer candidate keywords are favoured by deg(w)/freq(w). example .

Since candidate keywords are generated based on stop words. No candidate keyword will have any stop words in it (e.g. Times of India) . So to include them as candidate keywords, If pair of words occur twice in the document and in the same order then it they are added to candidate set of keywords.

RAKE's performance is evaluated against technical abstracts reported in Hulth (2003), and it achieved 33.7 % precision and 37.2 % recall with self generated stopwords (df > 10) which is higher than textrank's best score which is 31.2 % precision and 37.2 % recall.

2.2 Supervised Keyword Extraction Methods

2.2.1 KEA

KEA [9] describes about the keyphrase extraction and assignment. Keyphrase extraction and assignment are statistical learning methods requires a set of documents annotated with the manually assigned keywords.

keyphrase assignment is to select the phrases from the controlled phrase vocabulary that best describe a document. The training data mapped to each phrase in the vocabulary, separate classifier is learned for each phrase. A new (test) document is given to all classifiers, the phrase of classifier which gives maximum positive score is choosen. We are not further discussing this technique because it is less relevant to the current scenario, where meta learning does not fit to the controlled vocabulary learning.

keyphrase extraction is designed to choose the keyphrase from text document itself. It is based on lexical and information retrieval techniques to extract phrases from the document text. Training data is used to tune the parameters of each features.

Phrase Identification TODO

- Candidate phrases are limited to a certain maximum length (3 words).
- Candidate phrases are limited to a certain maximum length (3 words).
- Candidate phrases can not start or end with stop words.

All continous sequence of words in each sentence satisfy above three rules, are candidate phrases. Subphrases also part of candidate phrases.

Features

The initial version of KEA used only two features for deciding importance of phrases. They are TFxIDF and first occurence of each phrase in the document.

- **TFxIDF** TFxIDF is used as one of the feature. TF is the frequency of phrase in the test document and IDF is general usage or number documents the phrase used.

$$\mathrm{TFxIDF} = \frac{\mathrm{freq}(P, D)}{\mathrm{size}(D)} x - \log_2 \frac{\mathrm{df}(P)}{N}, where \tag{2.3}$$

  - freq(P,D) number of times phrase P occurs in document D.
  - size(D) is the number of words in D
  - df(P) is number of documents have the phrase P in total training corpus.
  - N is total number of documents in collection.

- **Positional Information** First occurence of phrase in the document is used as another feature. It is calculated by number words precede the phrase's first occurence divided by the number words in the document.

Both the features are discretized. The real valued features are divided by the range they fall into and assigned categorical values.

Classification

Each candidate keyphrase is classified into 'YES' or 'NO' which indicates whether the candidate phrase is important or not (keyphrase or not) based on feature values of phrases.

$$P[YES] = \frac{Y}{Y+N} P_{TFxIDF}[t|YES] * P_{DISTANCE}[d|YES] \tag{2.4}$$

Results

write

2.3 Image Retrieval

An image retrieval system is a system for browsing, searching and retrieving images from a large repository of digital images. Common methods of image retrieval utilize some method of adding metadata to images such as keywords or descriptions, so that retrieval can be performed over the annotation words.

To search for images, we/user need to provide query terms such as keyword or image file and the system is expected to return images 'similar' to the query.

Image retrieval system are broadly classified as,

- Image meta search
- Content based retrieval

2.3.1 Image meta search

Given query as words, and the descriptive meta data of each image is considered as text document. Image search system work as traditional information retrieval system where regardless of image semantics only description of image is only used for retrieval.

2.3.2 Content based retrieval

If we have documents and images annotated with them. Consider D is set of documents which contain both images and text. Images and texts are represented in term of feature vectors R^I and R^T respectively. These vectors represented in different vector space and there exist one-to-one mapping between them. Given text $T^q \in R^T$ we need to find an $I_q \in R_I$.

For finding an image based on text, we need to learn a mapping function

$$M : R^T - > R^I \tag{2.5}$$

Given text T^q it suffices to find nearest image $M(R^T)$. Since there is not direct correspondance between R^T and R^I. The mapping has to be learned from tranining sample. One way is to map each space into intermediate spaces U^T and U^I such they have correspondance.

$$M_I : R^I - > U^I \tag{2.6}$$

and

$$M_T : R^T - > U^T \tag{2.7}$$

The two isomorphic spaces U^I and U^T and there is invertible mapping

$$M : U^T - > U^I \tag{2.8}$$

Main problem is to find the subspaces U^T and U^I, one way is to find two linear projections

$$P_I : R^I - > U^I \tag{2.9}$$

and

$$P_T : R^T - > U^T \tag{2.10}$$

Correlation matching

Canonical Correlation Analysis (CCA) is a data analysis and dimensionality reduction method similar to Principle Component Analysis (PCA). Here, PCA deals with one dimension whereas CCA is joint dimensionality reduction of two e heterogeneous representations of the same data.

CCA and SCM - explain

Chapter 3

Experiments on Keyword Extraction

3.1 Unsupervised Approaches

3.1.1 Boosting based on frequency and co-reference

Simple method for the determining top keywords will be counting the occurrences of each word in the given document. However the problem with this method is Natural language text the word is represented in different forms. Stemming may help normalizing different verb representation with morphemes. But Images and news articles mostly centered around the entities and then verb as relations if required. *For example, Steve Jobs can be referred as Steve, Jobs, he, his, him... etc,.* When we want to find the fraction of sentences that does covers entities or nouns, we can use the co-reference resolution and anaphora resoultion. To identify the co-refered mentions in the article Stanford Co-reference Pipeline is used.

Steps of the experiments are,

1. Given document is tokenized in to lexical units.

2. All possible noun phrases from the parse tree are considered as candidate for final key phrases.

3. Stanford co-referencing pipleline ran on the sentences of articles. From the output of co-reference, the frequency of each noun phrase is calculated.

FREQUENCY BASED ON RAKING AFTER COREFERENCE

1. **EU** CHAIN27-["the EU 's" in sentence 2, "EU" in sentence 3, "The EU 's" in sentence 4, "EU" in sentence 6, "EU" in sentence 10, "EU" in sentence 11]
2. **European Commission** CHAIN19-["The European Commission , the EU 's executive ," in sentence 2, "The European Commission" in sentence 2, "European Commission" in sentence 2, "its" in sentence 2, "its" in sentence 2, "the Commission 's" in sentence 5, "the Commission" in sentence 6, "The Commission" in sentence 7, "the Commission" in sentence 8]
3. **duties** CHAIN7-["duties on billions of euros of Chinese solar panels" in sentence 1, "the duties" in sentence 7, "the duties" in sentence 8, "the duties" in sentence 10]

Different Weighting

Scoring methods like

- Giving more weightage to the mentions which appear in the first few sentences.
- Phrases in part of sentences,
- Referring part of phrase, considered to be talking about entity and boosted their scores. For example, in the below result, appearance of *duties* in *duties on billions of euros of Chinese solar panels* implicitly denotes we are talking about *chinese solar panels*. This needs to be studied with wide range of articles.

RANKING USING POSITIONAL WEIGTHING

1. **EU**
2. **European Commission**
3. **Chinese solar panels** :2:11.75: CHAIN13-["Chinese solar panels" in sentence 1, "solar panels" in sentence 11]
4. **billions** :2:9.5: CHAIN9-["billions of euros of Chinese solar panels" in sentence 1, "them" in sentence 2]
5. **duties** :4:8.0: CHAIN7-["duties on billions of euros of Chinese solar panels" in sentence 1, "the duties" in sentence "the duties" in sentence 8, "the duties" in sentence 10]

3.1.2 TextRank - Modified

Purpose is to tryout scoring methods other than frequency of entities. It is based on how surrounding entities are connected. We modified the TextRank [3], to take into account about co-refering mentions in the text. So, text graph is created in a different way than just considering co-occurence window.

Creation of Graph

- All the possible noun phrases the document used as nodes.
- Two consecutive nouns phrases in a sentences are connected. For example in the sentence ‘EU will impose anti-dumping levies.’, *EU* and *‘anti-dumping levies’* are connected in the graph.
- Two noun phrases/entities are connected, If they are co-referred. For example consider following two sentences ‘Ramu is a Student.’ and ‘He is intelligent.’, here *Ramu* and *He* are connected.

Above three steps create undirected graph with edges are unweighted. Page Rank or Random walk applied on this graph with damping factor 0.15,

$$WS(V_i) = (1 - d) + d * \sum_{j \in InV_i} \frac{W_{ji}}{\sum_{k \in OutV_j W_{jk}}} * WS(V_j) \tag{3.1}$$

RANKING USING MODIFIED TEXTRANK

1. **The European Commission** , the EU ’s executive, rank=1.4510955164654349, key=34, marked=true, sentNum=8, startWord=7, endWord=9, corefChain=19
2. **the EU**, rank=1.36762168208762, key=46, marked=true, sentNum=3, startWord=1, endWord=2, corefChain=28]
3. **duties on billions of euros of Chinese solar panels**, rank=1.3200079680210361, key=9, marked=true, sentNum=1, startWord=8, endWord=17, corefChain=8
4. **provisional duties**, rank=1.1529714795051218, key=42, marked=true, sentNum=11, startWord=5, endWord=7, corefChain=26]

3.2 Supervised Approaches

In the supervised approaches on keyword extraction, we consider the keywords are the words which appear in the meta data of image. We do remove the stop words of image meta data, all other words are consider to be keywords. and we try to learn a function which takes text document and title as input and produces the set of keywords which has maximal overlap with the keywords or metadata.

3.2.1 Naive Bayes

First one is using the Naive bayes classifier to find, the probability of being a word is 'keyword' or 'not'. We tokenized the text into sentences and words. Term frequency and Inverse Document Frequency is calculated after stemming the words.

titleScore is the probability of word being generated from title text. Often cases, title words are given importance, but difficulty is title is small in size so most of the cases the acronyms or shortened word. For example *united states* in text corresponds to *U.S.* in title, *mahindra singh doni* in text corresponds to *mahi* in title.

IDFTF score calculated based IR approaches.*Postag* for each word is obtained from Stanford PCFG parser.

Final score is calculated based on,

$P(Key/Word, PosTag, IDFTF, titleScore)$

$$= P(Key) * P(Word/Key) * P(PosTag/Key) * P(IDFTF/Key) * P(titleScore/Key) \tag{3.2}$$

Graphical model digram of Naive bayes approach is depicted in 3.1

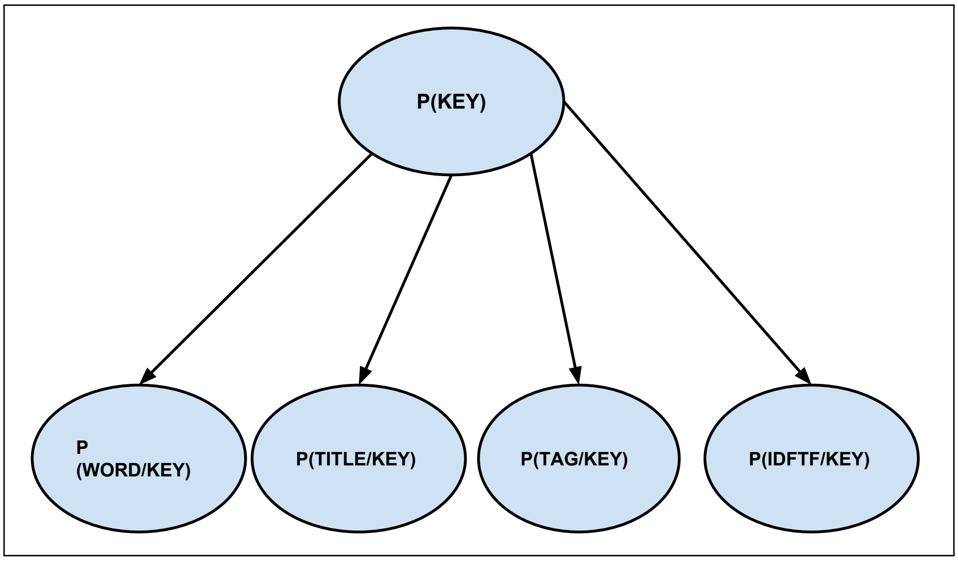

Figure 3.1: Naive Bayes Classification

Results: TODO

3.2.2 Hidden Markov Model

For this Hidden Markov Model, we use all the features we described in the previous section. Here, we try to label each sentence as {YES, NO}. This method is different from the earlier method in a way that we consider state transition probabilities are considered. It is likely to perform better.

The Best key sequence is

$$KeySeq^* = argmax_{KeySeq} P(KeySeq/Word, PosTag, IDFTF, titleScore) \tag{3.3}$$

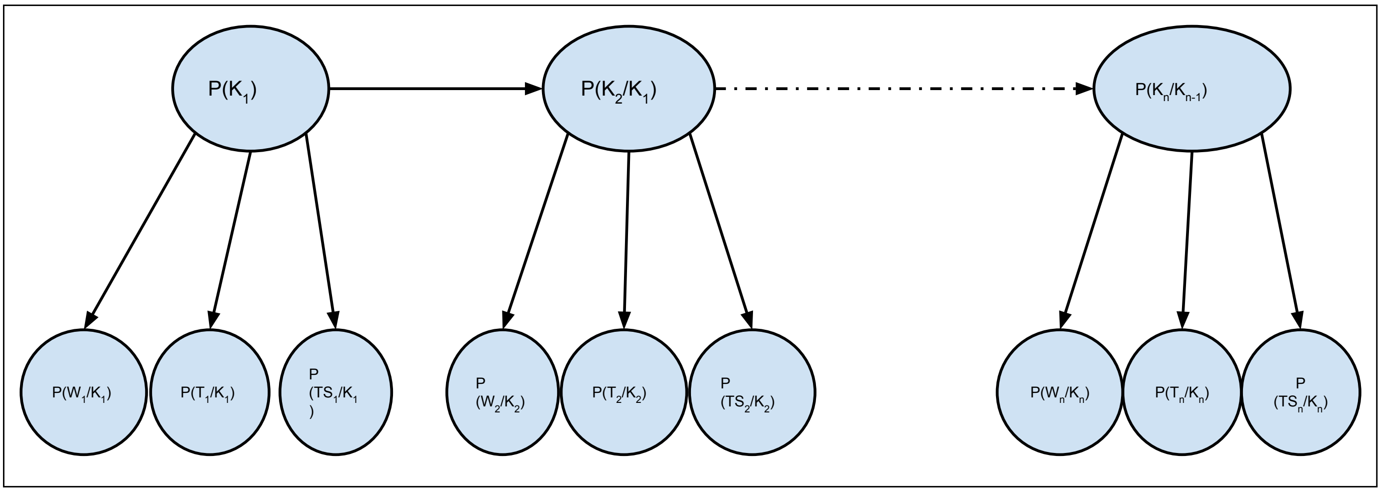

Figure 3.2: Hidden Markov Model

$$P(KeySeq/Word, PosTag, IDFTF, titleScore)$$

$$=P(KeySeq) * P(Word, PosTag, IDFTF, titleScore/KeySeq) \tag{3.4}$$

Using chain rule,

$$=P(KeySeq) * P(Word/KeySeq) * P(PosTag/KeySeq, Word)$$
$$* P(IDFTF/KeySeq, Word, PosTag) * P(titleScore/KeySeq, KeySeq, Word, PosTag, IDFTF) \tag{3.5}$$

Independence assumptions,

$$=P(KeySeq) * P(Word/KeySeq) * P(PosTag/KeySeq)P(IDFTF/KeySeq) * P(titleScore/KeySeq) \tag{3.6}$$

Using markov assumption,

$$\begin{aligned}=&P(Key_1) * P(Key_2/Key_1) * .. * P(Key_n/key_{n-1})*\\ &P(Word_1/Key_1) * P(Word_2/Key_2) * .. * P(Word_n/Key_n)*\\ &P(PosTag_1/Key_1) * P(PosTag_2/Key_2) * .. * P(PosTag_n/Key_n)*\\ &P(IDFTF_1/Key_1) * P(IDFTF_2/Key_2) * .. * P(IDFTF_n/Key_n)*\\ &P(titleScore_1/Key_1) * P(titleScore_2/Key_2) * .. * P(titleScore_n/Key_n)\end{aligned} \tag{3.7}$$

Results : TODO

Chapter 4

Conclusion and Future work

In this work, we have studied various ways of finding relation between large text and short text through extracting keyword for the purpose of image retrieval for the article.

And experimented with unsupervised and supervised approaches. However, nature of problem requires further understanding and improved methods.

In future, we are planning to using the external resources to better understand the relation between the large text and short text. For example, using the *click-through logs* will be helpful.

4.1 Future Plan

4.1.1 Short term

Short term plan is to complete the tagging of atleast one classification of images. *Representative images* are the one, which are easy relatively compared to other types. and Improvement on current approaches based on error analysis.

4.1.2 Medium term

4.1.3 Long term

Long term goal is to complete full working system, with using external resources apart from a given news article and descriptive meta data,

1. Utilizing the click through logs and query words.

2. Utilizing the semantic data from Freebase, or others.
3. Feedback based learning
4. Clustering as approach to make a bottom-line of current news article based articles appeared previous day or week.

References

[1] Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 28–34. Association for Computational Linguistics, 2011.

[2] Yutaka Matsuo and Mitsuru Ishizuka. Keyword extraction from a single document using word co-occurrence statistical information. *International Journal on Artificial Intelligence Tools*, 13(01):157–169, 2004.

[3] Rada Mihalcea and Paul Tarau. Textrank: Bringing order into texts. In *Proceedings of EMNLP*, volume 4. Barcelona, Spain, 2004.

[4] Amit Kumar Mondal and Dipak Kumar Maji. Improved algorithms for keyword extraction and headline generation from unstructured text.

[5] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: bringing order to the web. 1999.

[6] Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. Automatic keyword extraction from individual documents. *Text Mining*, pages 1–20, 2010.

[7] Yasin Uzun. Keyword extraction using naïve bayes. *Bilkent University, Department of Computer Science*, 2005.

[8] Christian Wartena, Rogier Brussee, and Wout Slakhorst. Keyword extraction using word co-occurrence. In *Database and Expert Systems Applications (DEXA), 2010 Workshop on*, pages 54–58. IEEE, 2010.

[9] Ian H Witten, Gordon W Paynter, Eibe Frank, Carl Gutwin, and Craig G Nevill-Manning. Kea: Practical automatic keyphrase extraction. In *Proceedings of the fourth ACM conference on Digital libraries*, pages 254–255. ACM, 1999.