Pattern Based Keyword Extraction for Contextual Advertising

Kushal Dave
Language Technologies Research Centre
International Institute of Information Technology
Hyderabad, India
kushal.dave@research.iiit.ac.in

Vasudeva Varma
Language Technologies Research Centre
International Institute of Information Technology
Hyderabad, India
vv@iiit.ac.in

ABSTRACT
Contextual Advertising (CA) refers to the placement of ads that are contextually related to the web page content. The science of CA deals with the task of finding advertising keywords from web pages. We present a different candidate selection method to extract advertising keywords from a web page. This method makes use of Part-of-Speech (POS) patterns that restrict the number of potential candidates a classifier has to handle. It fetches words/phrases that belong to the selected set of POS patterns. We design four systems based on chunking method and the features they use. These systems are trained on a naïve Bayes classifier with a set of web pages annotated with ‘advertising’ keywords. The systems can then find advertising keywords from previously unseen web pages. Empirical evaluation shows that systems using the proposed chunking method perform better than the systems using N-Gram based chunking. All improvements in the systems are found statistically significant at a 99% confidence interval.

Categories and Subject Descriptors
H.3.1 [Content Analysis and Indexing]: Abstracting, Linguistic processing; H.4.m [Information Systems]: Miscellaneous

General Terms
Algorithm, performance, experimentation

Keywords
Keyword Extraction, Contextual Advertising

1. INTRODUCTION
Online advertising is fast becoming one of the most popular means of reaching millions of audiences in a single advertisement effort. Placing contextually related ads has a twofold advantage: First, these ads are less annoying to the user and second, it also increases the probability of user clicking on the ads. Placing ads matching the context of a web page involves selecting ads from the ad database based on the web page content. Typically, this is done by first finding advertising keywords from a web page, in some cases expanding these keywords and retrieving a set of ads based on these expanded keywords. Hence, keyword extraction is one of the fundamental tasks for most of the contextual advertising systems. Keyword extraction in the web setting becomes more difficult in comparison to domain specific keyword extraction because of the diverse data and presence of noisy text (navigational blocks, copyright and privacy notes etc.) on the web pages.

Most of the keyword extraction systems [2, 6, 7, 8] first tokenize the text into all possible words or phrases (N-Grams). This process of tokenization is called chunking and the token word/phrase is called candidate. A classifier then classifies each candidate, based on some features, into a keyword or a non-keyword.

In this paper we propose the following: (1) A new chunking approach, that employs POS knowledge for selecting candidates. (2) We try exploiting the linguistic context of the advertising keywords by using POS tag of the candidates and POS tag of words surrounding them as features. (3) This system with the POS chunking method performs substantially better than the N-Gram approach used in some earlier advertising keyword extraction systems [7, 8].

2. SYSTEM ARCHITECTURE
This section describes various components used by the system: preprocessor, candidate selector, classifier & features.

2.1 Preprocessor
The first task in the preprocessing of web pages is to clean the extraneous content from the web pages. After cleanup, the useful content comprises the title and body of a web page. This is done so that the annotators only get a snapshot of web page and the annotation is not biased due to the extraneous content. We apply POS tagging to the cleaned text (navigational blocks, copyright and privacy notes etc.) on the web pages.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 2010 ACM 978-1-4503-0099-5/10/10 ...$10.00.
2.2.2 POS Chunking

Even after keeping the sentence boundary constraint, it was found that many of the candidates generated using the N-gram chunking method do not qualify as a phrase and hence are not potential advertising keywords. For Example in the sentence:

‘Nikon unveils eight new Coolpix cameras’.

N-gram chunking would result in many non-advertising keywords like:


‘Nikon unveils eight’, ‘unveils eight new’, etc.

In all, N-Gram chunking selects 19 candidates out of which only four are advertising keywords. Having so many trivial non-advertising phrases in the set of candidates increases the processing time of the system, as stated in Yih et al. [8]. This also affects the accuracy of the system as the classifier may mistakenly classify one of these as an advertising keyword. Such trivial non-advertising keywords can be reduced to a great extent by using POS knowledge.

Hulth [3] employed a similar approach for general keyword extraction. The POS chunking approach takes advantage of the fact that the advertising keywords are proper phrases in that language. Hence only these potential advertising keywords are considered as candidates. These proper phrases usually follow certain POS patterns. These POS patterns can be learned from the manually annotated data. One way to learn from the data is to pick the most frequent patterns from the data. In our case, we found that the frequent patterns contained one or more noun tags (NN, NNP, NNS and NNPS) along with adjective tags (JJ) and in some cases cardinal tags (CD). Following are some of the frequent patterns followed by the ad keywords:

- (Noun): For keywords containing one or more noun tags. For Example, Sony PlayStation, Nokia N95 mobile, etc.
- (Adjective) (Noun): For keywords starting with an adjective followed by one or more nouns. For Example, digital cameras, flat screen TV, etc.
- [CD] (Noun) / (Noun)[CD]: For keywords containing one or more nouns and a cardinal. For Example: Microsoft Xbox 360, 3 Mobile, etc.
- [DT] (Noun): For keywords such as movie name, book name etc. For Example: The Spiderman etc.

Here, + indicates one or more occurrences of that POS tag. Tags occurring in square brackets are optional.

Chunking based on these patterns result in reducing the candidate set to a large extent. These reductions get rid of non-advertising keywords that are present in the set of candidates. For the sentence described above, POS chunking would only select following six candidates:


Thus, for a sentence of length six words, 13 non-advertising candidates are reduced using POS chunking method. As we show in the Section 4.1, The average reduction in the number of candidates is found to be 80% in comparison with N-Gram approach.

We only consider candidates up to length six, because very few phrases of length greater than six follow the patterns. Among the candidates of length greater than six, negligible number of candidates stood as advertising keywords.

2.3 Classifier

The classifier we use is a binary classifier that classifies a keyword as an advertising keyword or a non-advertising keyword. The classifier is trained with manually annotated examples. We tried many learning algorithms such as naïve Bayes, logistic regression and bagging. For the kind of features we have, naïve Bayes outperforms other learning classifiers for both the N-gram and POS chunking approach. Also naïve Bayes trains faster than other learning algorithms. Hence naïve Bayes is used as a learning algorithm for the classifier. The candidates are ranked based on the probability of the candidate being an advertising keyword.

2.4 Features

Choosing a good set of feature is very important as it largely affects the performance of a classifier [4]. We describe the following three categories of features:

2.4.1 Linguistic Features (LING)

These features represent the linguistic context of a candidate by taking into consideration POS tag of the candidate and its surrounding words. For Example, after training, the classifier might learn that most of the candidates are prefixed by a determiner or they occur more at the start of a sentence etc. Following are the features in this category:

- CurrentPOS: The POS tag of the candidate.
- SingleFPOS: The POS tag of the word occurring immediately to the left of the candidate.
- DoubleFPOS: The POS tag of two words occurring immediately to the left of the candidate.
- SingleBPOS: The POS tag of the word occurring immediately to the right of the candidate.
- DoubleBPOS: The POS tag of two words occurring immediately to the right of the candidate.

All the features in these category are nominal features.

If a candidate occurs at multiple places in the web page, then the feature value assigned to it is the most frequently occurring POS tag. In case of a tie, the tag occurring first is given preference. Assigning different tags to the same word/phrase at different locations can happen due to the error in POS tagger.

While fetching the POS tag of the word surrounding the candidate, the sentence boundary was taken into consideration. For e.g. if the candidate is the first word in a sentence then its SingleFPOS and DoubleFPOS feature values will be null. There were a lot of potential advertising keywords surrounded by a comma, hence comma was not considered as a sentence boundary, instead it was given an additional POS tag COMMA.

2.4.2 IR Oriented Features (IR)

We use term frequency (TF) of the candidate as a feature. More the number of times a word/phrase occurs, better are the chances of it being a keyword. For the POS systems, while calculating TF both the candidate and the content are stemmed. This is done, so that both ‘Video games’ and ‘Video game’ are treated as the same keyword. Along with TF, we use its natural logarithm log(TF) as a feature.

2.4.3 Other Features (OF)

All these features are binary features (0 or 1) apart from length, which is a numeric feature. In this category features like whether the candidate occurs in title, whether it is present in the anchor text, whether it is capitalized, does
it occur frequently in the query log etc are used. In order to incorporate the query feature, AOL and Yahoo! query logs were used. AOL query log consisted of ≈20M web queries collected from ≈650k users over a period of three months. We removed infrequent (occurring less than three times) queries from this AOL query log. We also added the Yahoo! query log which contained the 1000 most frequent queries issued to the Yahoo! search engine.

3. EXPERIMENTS

3.1 Dataset & System Designs

The dataset comprised web pages in English from various categories such as blog, product review pages, forums and news articles. We concentrated on web pages where contextual advertising seemed desirable, that is, we only took pages which had advertisements placed by some Ad-network. In order to maintain the diversity of the dataset we took pages that talked about movies, electronic gadgets, books, songs, computers, games, computer accessories etc. Category wise distribution of the web pages is as shown in Table 1.

<table>
<thead>
<tr>
<th>Table 1: Category wise distribution of the dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete Dataset</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>810</td>
</tr>
</tbody>
</table>

We designed four systems as shown in Table 2. It shows the chunking approach and the set of features used by a system. For Example, POS+ling system uses POS chunking and ling, IR and OF features. All the four systems use the naïve Bayes classifier. We designed these four systems in order to assess: (1) Performance improvement in POS chunking method over N-Gram chunking method. (2) Whether Linguistic features (ling) help in improving the precision of the system. (3) How does POS chunking method perform in conjunction with the linguistic features. (POS+ling).

<table>
<thead>
<tr>
<th>Table 2: Description of the systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>System</td>
</tr>
<tr>
<td>--------------------------------------</td>
</tr>
<tr>
<td>POS+ling</td>
</tr>
<tr>
<td>POS-ling</td>
</tr>
<tr>
<td>N-Gram+ling</td>
</tr>
<tr>
<td>N-Gram-ling</td>
</tr>
</tbody>
</table>

In Section 4.2.3, we compare the performance of POS systems with the Keyword Extraction Algorithm (KEA) [2]. As the data we are dealing with is diverse, KEA is implemented without any domain knowledge. KEA employs the N-Gram chunking approach and uses term frequency, position of the first occurrence of candidate in the page and length of the candidate as features.

3.2 Performance Measures

To check the efficiency of the POS approach, we calculated its coverage. By coverage, we mean the percentage of advertising keywords that are chosen in the candidate set amongst all advertising keywords on that web page.

We used the average precision at Nth position (P@N) to evaluate performance of the systems. We test the precision at first (P@1), third (P@3) and fifth (P@5) positions.

We also performed significance testing for the results presented in this paper using paired t-test as suggested by Sanderson and Zobel [5]. We considered N-Gram systems as a baseline and measured the improvement in POS systems at 99% confidence interval i.e at p-value less than 0.01.

For POS+ling the baseline system was N-Gram+ling, while for POS-ling the baseline was N-Gram-ling.

<table>
<thead>
<tr>
<th>Table 3: Coverage values for the complete dataset and its categories for POS chunking method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete Dataset</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>Coverage</td>
</tr>
</tbody>
</table>

3.3 Training and Testing

For training and testing purpose, the 810 web pages were manually annotated. The pages were annotated by fellow researchers from the lab. They were explained how a contextual advertising system works, and how it places ads based on the extracted list of keywords. The annotators were also shown example web pages where contextual advertisements were placed. They were given several examples such as: “If a web page talks about graphic cards then Nvidia might want to place their ads on the web page. Moreover, they were asked to label each reference of an advertising keyword occurring on the page. For Example, if ‘Sony Playstation 3’ and ‘PS3’ both occurred then they were asked to label both the keywords. For inter-annotator agreement, the first 30 pages were tagged by all of the annotators and these annotations were discussed with the annotators for uniformity.

We perform a five-fold cross validation with the 810 page dataset, that is using 80% data (648 web pages) for training and 20% (162 web pages) for testing for each fold.

4. RESULTS

4.1 Chunking

After running POS chunking on our dataset, the average reduction in total number of candidates was found to be 86%. A reduction of this magnitude improves the performance of the system significantly, as the classifier will have that much lesser candidates to deal with. As shown in Table 3, coverage for complete dataset was found to be 93%. The remaining 7% advertising keywords were missing due to the errors in POS tagging or because few advertising keywords did not follow the POS patterns (see Section 5). Coverage values for N-Gram chunking, intuitively, is 100%.

4.2 Performance of Systems

4.2.1 Complete Dataset

In this section, we evaluate the contribution of the chunking methods and the linguistic features. Table 5 shows performance of N-Gram and POS systems with (+ling) and without linguistic (-ling) features. The precision values are averaged over the five folds. Both versions of POS system (+ling & -ling) outperform their counterpart N-Gram systems. Following can be inferred from the results in Table 5:

- The only difference between a POS+ling system and an N-Gram+ling system is the chunking method. Hence, we infer that the improvement in precision in POS+ling system is because of the POS chunking method. A similar inference can be drawn by comparing results of POS+ling system with N-Gram+ling system.
- Similarly, after comparing POS+ling with POS-ling system and N-Gram+ling with N-Gram-ling, we infer that the slight improvement in precision due to the use of linguistic features.
- POS+ling system and N-Gram+ling system differ both in chunking method and the linguistic features. Hence we state that POS chunking and linguistic feature together boost the performance of the system.

---

1Yahoo! Webscope (Yahoo! Search Query Logs for English - LS), http://research.yahoo.com/Academic_Relations
Table 4: Performance of all the systems on individual categories of web pages (p-value ≤ 0.01)

<table>
<thead>
<tr>
<th>System</th>
<th>Blog</th>
<th>Product</th>
<th>News</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P@1</td>
<td>P@3</td>
<td>P@5</td>
</tr>
<tr>
<td>POS+ling</td>
<td>0.50</td>
<td>0.49</td>
<td>0.49</td>
</tr>
<tr>
<td>Ng+ling</td>
<td>0.39</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>POS-ling</td>
<td>0.49</td>
<td>0.46</td>
<td>0.46</td>
</tr>
<tr>
<td>NG-ling</td>
<td>0.35</td>
<td>0.35</td>
<td>0.37</td>
</tr>
</tbody>
</table>

Table 5: Performance of the N-Gram and POS systems (p-value ≤ 0.01)

<table>
<thead>
<tr>
<th>System</th>
<th>P@1</th>
<th>P@3</th>
<th>P@5</th>
</tr>
</thead>
<tbody>
<tr>
<td>POS+ling</td>
<td>0.47</td>
<td>0.48</td>
<td>0.46</td>
</tr>
<tr>
<td>NGram+ling</td>
<td>0.38</td>
<td>0.40</td>
<td>0.40</td>
</tr>
<tr>
<td>POS-ling</td>
<td>0.45</td>
<td>0.47</td>
<td>0.45</td>
</tr>
<tr>
<td>NGram-ling</td>
<td>0.35</td>
<td>0.38</td>
<td>0.38</td>
</tr>
</tbody>
</table>

4.2.2 Individual Categories

We tested the performance of all four systems on individual categories of the dataset, the results are as shown in Table 4. We could not perform the experiments for forums as their percentage in the dataset was less. The model used for testing was a single category of web page was trained using pages from the same category only. Product web pages give the highest precision amongst all the categories, as these pages have a lot of advertising keywords. Precision values for blogs and news on average are good. As shown, the POS+ling system performs better than all other systems for all categories of the dataset, for all precision values. These results complement the results described in section 4.2.1.

Table 6: Performance of POS+ling against KEA (p-value ≤ 0.01)

<table>
<thead>
<tr>
<th>System</th>
<th>P@N</th>
<th>POS+ling</th>
<th>KEA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P@1</td>
<td>0.38</td>
<td></td>
</tr>
<tr>
<td></td>
<td>P@3</td>
<td>0.48</td>
<td>0.44</td>
</tr>
<tr>
<td></td>
<td>P@5</td>
<td>0.46</td>
<td>0.43</td>
</tr>
</tbody>
</table>

4.2.3 KEA v/s POS+ling

We compare our best performing system i.e. POS+ling with KEA [2]. Same training and testing set were used for both the systems. The result are shown in Table 6. The POS+ling system performs significantly better than KEA for all precision values.

4.3 Significance Testing

To evaluate the statistical significance of the improvement in POS system over a N-Gram system, we conducted a paired t-test experiment between POS+ling and NGram+ling systems for all the precision values over a set of iterations. For each fold, we randomly split our test set of 162 pages into two disjoint and independent sets of 81 pages each. For each set, we find the p-value by comparing the POS+ling and NGram+ling system. This experiment was repeated 50 times to ensure the results are stable. The p-values are averaged over these 50 iterations for both Set-1 and Set-2 and across all the folds. Dividing the test dataset into two disjoint set helps removing any bias in the set selection. As can be seen from Table 7, the average p-values over these iterations are less than or equal to 0.01 which indicates statistical significance.

5. DISCUSSION & CONCLUSION

As explained in Section 4.1, about 7% keywords are missed either because they skipped the frequent POS patterns or due to POS tagging error. For e.g. On a movie web page the keyword 'Harry Potter and the Half-Blood Prince' is an advertising keyword, but it does not follow any of the POS patterns mentioned above. Such cases can be easily handled by removing stop words from the ad while matching against it. Now keywords 'Harry Potter' & 'Half-Blood Prince' can be matched with the ad text without function words. Also, some keywords were missing due to the POS tagging error.

The improvement in the precision of POS systems incurs a cost of tagging the content. If the web page is available before-hand then tagging information can be saved prior to chunking, without requiring any extra time for tagging. In cases where the web page content is available only on run time, the POS tagger will have to be used online.

POS systems for other languages can be implemented by using taggers for the specific language. The Stanford POS tagger that we use provides support for Chinese, German and Arabic languages. There are state-of-the-art POS taggers available for other European languages like (Swedish, Polish) and Asian languages like Hindi, Marathi etc.

The systems using the proposed POS chunking method showed improved precision compared to the systems using the N-Gram chunking. The linguistic features also contribute to the accuracy of the system. Also, the use of POS chunking method and the linguistic features together can improve the performance to a large extent. We also compared our best performing system (POS+ling) with few of the baseline keyword extraction systems such as KEA and system (NGram+ling system) described by Yih et al. [8] and showed that it performs better than both the systems.

6. REFERENCES