Maximum Entropy Context Models for Ranking Biographical Answers to Open-Domain Definition Questions

Alejandro Figueroa
Yahoo! Research Latin America,
Av. Blanco Encalada 2120,
4th floor, Santiago, Chile
afiguero@yahoo-inc.com

John Atkinson
Department of Computer Sciences
Universidad de Concepción
Concepción, Chile
atkinson@inf.udec.cl

Abstract

In the context of question-answering systems, there are several strategies for scoring candidate answers to definition queries including centroid vectors, bi-term and context language models. These techniques use only positive examples (i.e., descriptions) when building their models. In this work, a maximum entropy based extension is proposed for context language models so as to account for regularities across non-descriptions mined from web-snippets. Experiments show that this extension outperforms other strategies increasing the precision of the top five ranked answers by more than 5%. Results suggest that web-snippets are a cost-efficient source of non-descriptions, and that some relationships extracted from dependency trees are effective to mine for candidate answer sentences.

Introduction

Generally speaking, definition questions are found in the form of strings like "*What is a <concept>?*" and "*What does <concept> mean?*". This specific class of query covers indeed more than 20% of the inputs within query logs, hence their research relevance (Rose and Levinson 2004).

Unlike other kinds of question types, definition questions expect a list of pieces of information (nuggets) about the concept being defined (a.k.a. the *definiendum*) as an answer. More precisely, the response is composed of, but not exclusively, of relevant biographical facts. A question-answering (QA) system must therefore process several documents so as to uncover this collection of nuggets. To illustrate this, a good response to the question "*What is ZDF?*" would involve -sentences embodying- facts such as "*Second German Television*", "*public service*" and "*based in Mainz*".

A general view of the question-answering process points to a pipeline commonly composed of the following steps: candidate answer retrieval, ranking, selection and summarisation. In the first step, candidate answers are fetched from a target corpus, and singled out by some definiendum matching technique and/or a fixed set of definition patterns. The second phase typically involves a scoring function based on the accuracy of the previous alignments (H. Joho and M. Sanderson 2000; 2001), keywords learnt from web-snippets and/or knowledge base (KB) articles such as $Wikipedia$ (Katz et al. 2007), *Merriam-Webster* dictionary (Hildebrandt, Katz, and Lin 2004), and $WordNet$ (Echihabi et al. 2003; Wu et al. 2005). The selection stage entails an experimental threshold that cuts off candidate answers, and the summarisation applies a redundancy removal strategy.

Nowadays, there are two promising trends for scoring methods: one is based on *Language Models* (LMs) which mainly rates biographical[1] answers (Chen, Zhon, and Wang 2006), whereas the other is based on discriminant models which distinguishes short general descriptions (Androutsopoulos and Galanis 2005; Lampouras and Androutsopoulos 2009).

Related Work

There are numerous techniques designed to cope with definition queries. One of the most prominent involves the extraction of nuggets from KBs, and their further projection into the set of candidate answers (Cui, Kan, and Xiao 2004; Sacaleanu, Neumann, and Spurk 2008). More specifically, these nuggets are used for learning frequencies of words that correlate with the definiendum, in which a centroid vector is formed so that sentences can be scored according to their cosine distance to this vector. The performance of this kind of strategy, however, falls into a steep drop when there is not enough coverage for the definiendum across KBs (Zhang et al. 2005; Han, Song, and Rim 2006). In other words, it fails to capture correct answers verbalised with words having low correlation with the definiendum across KBs, generating a less diverse outcome and so decreasing the coverage.

In general, centroid vector-based approaches rate candidate answers in congruence with the degree in which their respective words typify the definiendum. The underlying principle is known as the Distributional Hypothesis (Harris 1954; Firth 1957) in which KBs yield reliable characterising terms. An additional aspect that makes this method less attractive is that term co-occurrences do not necessarily guarantee a meaningful syntactic dependency, causing the selection of manifold spurious answers.

In order to address this issue, (Chen, Zhon, and Wang

[1] The term "biographical", in a broader sense, is used as a synonym of content found in encyclopedias for different sorts of definienda such as companies and countries.

2006) extended the centroid vector based method to include word dependencies. First, they learn frequent stemmed co-occurring terms derived from top-ranked web snippets, which were fetched via a purpose-built query reformulation method. By retaining their original order, these words are then used for building an ordered centroid vector representation of the sentences, wherewith unigram, bigram and biterm LMs were constructed. Experiments indicate that biterm LMs significantly improve the performance in relation to the original centroid vector method. Thus, the flexibility and relative position of lexical terms are observed to encapsulate shallow information about their syntactic relation (Belkin and Goldsmith 2002).

A related work (Figueroa and Atkinson 2009) built contextual models to tackle the narrow coverage provided by KBs. Unlike previous methods, context models mine sentences from all $Wikipedia$ pages that align the pre-defined rules in table 1. These matched sentences are then clustered in accordance with their context indicator (e.g., "*author*", "*player*" and "*song*"), which is generally given by the root of the dependency tree:

author:
CONCEPT is an accomplished author.
CONCEPT, a bestselling childrens author.
player:
CONCEPT is a former ice hockey player.
CONCEPT, a jazz trumpet player.
song:
CONCEPT, the title of a song for voice and piano.
CONCEPT is a rap song about death.

Next, an n-gram ($n = 5$) LM is constructed for each context, in which unseen instances bearing the same context indicator are rated. This constituted a key difference to earlier techniques, which predicate largely on knowledge regarding each particular definiendum found across KBs. Another advantage of context models is their bias in favour of candidate answers carrying more relevant indicators across both KBs and candidate answers (e.g., "*band*" in the event of the definiendum "*The Rolling Stones*"). This method exploits contextual semantic and syntactic similarities across lexicalised dependency trees of matched sentences. As a result, context models cooperate on improving precision and ranking with respect to bi-term LMs.

One common drawback between previous strategies (Cui, Kan, and Xiao 2004; Chen, Zhon, and Wang 2006; Figueroa and Atkinson 2009) arises from the absence of information about non-descriptions, accounting solely for positive samples. This has an impact on the ranking as many words, bi-terms or dependency paths that are predominant in definitions can also appear within non-descriptions (e.g. *band→metal* in "*definiendum is a great metal band.*").

As for discriminant models for definition ranking, maximum entropy models have been preferred as (Fahmi and Bouma 2006) showed that for a language different from English they achieve good performance. Other QA methods (Miliaraki and Androutsopoulos 2004; Androutsopoulos and Galanis 2005) have also been promising to score 250-characters open-domain general definitions using a Support Vector Machine (SVM) trained with mostly surface attributes extracted from a web corpus. In addition, SVM classifiers have also been exploited with surface features to rank sentences and paragraphs about technical terms (Xu et al. 2005). Incidentally, (Androutsopoulos and Galanis 2005; Lampouras and Androutsopoulos 2009) automatically gathered and annotated training material from the Internet, whereas (Xu et al. 2005) manually tagged a corpus originated from an Intranet. Nevertheless, these techniques do not benefit from context models.

Maximum Entropy Context Models for Definitional Questions

In a nutshell, our work extends context models to account for regularities across non-descriptions, which are collected from sentences extracted from web-snippets. This collection of sentences is limited in size and takes advantage of context models splitting the positive data into small training sets. A portion of these web sentences was manually labeled so as to obtain non-descriptions, while an extra proportion of negative samples was automatically tagged by a LM built on top of these manually annotated samples. Finally, a Maximum Entropy (ME) Model is generated for each context, wherewith unseen testing instances of candidate answers are rated.

Corpus Acquisition

In our approach, negative and positive training sets are extracted differently. The former was acquired entirely from the Web (i.e., web snippets), while the latter came from $Wikipedia$ and web snippets.

This web training data is obtained by exploiting a definition QA system operating on web-snippets (Figueroa 2008). In order to generate the final outcome, the model takes advantage of conventional properties such as word correlations, and the manually-built definition patterns shown in table 1, and redundancy removal tasks. The average F(3)-score of the model is 0.51 on a small development set, and this system ran for more than five million definienda originated from a combination of $Wikipedia$ and $FreeBase$[2], randomly selected. This model collects a group of diverse and unlabelled web snippets bearing lexical ambiguities with genuine definitions, which would discard "easy-to-detect" non-descriptions. Overall, this corpus involves about 23,500,000 web snippets concerning about 3,700,000 different definienda, for which at least one sentence was produced by the system. Note that web-snippets were preferred to full-documents in order to avoid their costly processing, and due to the fact that they convey localised context about the definiendum. The average length of sentences mined from web-snippets was 125 characters.

Extracting Positive Examples

First of all, unlike previous methods (Xu et al. 2005; Androutsopoulos and Galanis 2005; Fahmi and Bouma 2006; Lampouras and Androutsopoulos 2009), entries from $Wikipedia$ were taken into consideration when acquiring a positive training set. These are then split into sentences

[2] http://www.freebase.com/

Surface Patterns	Example
1. δ [is|are|has been|have been|was|were] [a|the|an] ρ	Ayyavazhi is a dharmic belief system which originated in...
2. δ, [a|an|the] ρ [,|.]	Barrack Hussein Obama, the US President, ...
3. δ [become|became|becomes] ρ	Ibn Abbas became blind during his last years, ...
4. δ [|,] [which|that|who] ρ	Eid al-Adha , which occurs at the end of the month of....
5. δ [was born] ρ	Alan George Lafley was born on June 13, 1947, in Keene,...
6. δ, or ρ	Tokyo Tsushin Kogyo, or Totsuko,...
7. δ [|,][|also|is|are] [called|named|known as] ρ	Abd al-Aziz ibn Saud, known as Ibn Saud, a descendant of...
8. δ (ρ)	Aesop (from the Greek Aisopos), famous for his fables, ...

Table 1: Surface patterns (δ and ρ stands for the definiendum and the description, respectively).

Web-sites	
encarta.msn.com	www.rootsweb.ancestry.com
www.fact-index.com	encycl.opentopia.com
en.allexperts.com	www.absoluteastronomy.com
www.zoominfo.com	encyclopedia.farlex.com
wikitravel.org	www.1911encyclopedia.org
www.britannica.com	commons.wikimedia.org
www.reference.com	www.probertencyclopaedia.com

Table 2: List of manually chosen authoritative hosts (positive samples).

which are clustered according to their context indicators as shown in the example in the related work. The result involves about three million different sentences over 55,000 distinct context indicators.

Unlike the original context models, the previous categorisation allowed multi-contexts. To clarify this, consider the phrase "*definiendum was the author and illustrator of many children's books.*" Following the original approach, this sentence was included in the cluster "*author*", because this is the main context. However, this was also incorporated into the group "*illustrator*", that is provided by a secondary context, which is identified by performing two tasks:

1. Examining coordinations of nouns using the main context indicator.
2. Checking whether the remaining nouns within these coordinations exist in the previous set of 55,000 main context indicators. It allows us to deal with data-sparseness, and at the same time, to keep relevant nouns from the list.

Secondly, sampled short descriptions, such as "*definiendum is a non-profit organization.*", were selected from the web corpus. For this, they must meet the following criteria:

- Having a frequency higher than four when exact matching is performed;
- Belonging to two distinct authoritative hosts (see table 2) or four different hosts. This list of dependable hosts was manually compiled by inspecting the most frequent references across this web corpus.

Eventually, this contributed with over 47,000 new different sentences pertaining to almost 4,000 distinct context indicators. These instances were added to the previous $\approx$55,000 clusters accordingly.

Nevertheless, these succinct descriptions can also be exploited for chosing extra and longer positive instances from our web-snippets corpus. Specifically, it was carried out by ensuring that these additional samples start with any of almost 47,000 prior reliable brief descriptions which were manually annotated. Thus, almost 195,000 different positive samples were extracted with respect to ca. 3,700 distinct context indicators, and about 52,500 distinct negative samples concerning $\approx$3,500 different context indicators.

As for the annotation process, examples were considered positive only if they were entire non-truncated descriptions. In other words, negative samples involve short truncated phrases such as:

definiendum is a book published
definiendum is a song officially released
definiendum is an airport opened

Also negatives samples comprised sentences that put descriptive and non-descriptive information together, such as follows:

definiendum is a comic book especially suited for our time.
definiendum is the sort of player needed to win the big games.
definiendum is the best known candidate.

Note that full non-descriptive sentences were also tagged as negative. Partial/truncated elements were seen as negative whenever the descriptive portion is prominent in the corresponding positive context (e.g., "*a comic book*"), which prevents the positive set from populating with non-descriptive/truncated data while keeping the descriptive information more representative of the positive class. The objective is to reduce the importance of relevant brief phrases (e.g., "*a comic book*"), which presence can lead candidate answers to rank at the top. This aims for full descriptions to rank higher than partial or truncated testing instances.

A placeholder was used for smoothing data-sparseness, more precisely, for substituting named entities and tokens starting with a capital letter or a number. Here, named entities were identified via Named Entity Recognition (NER) tools[3]. The lexicalised dependency trees of this corpus were then obtained by using a Dependency Parser[4]. All in all,

[3] http://nlp.stanford.edu/software/CRF-NER.shtml.

[4] http://nlp.stanford.edu/software/lex-parser.shtml.

Web-sites	
www.telegraph.co.uk	www.imdb.com
www.fullbooks.com	www.flickr.com
www.tribuneindia.com	www.bbc.co.uk
www.washingtonpost.com	www.tv.com
sports.espn.go.com	www.scribd.com
www.angelfire.com	www.amazon.com
www.tripadvisor.com	www.spock.com
www.geocities.com	www.nytimes.com
www.guardian.co.uk	www.myspace.com
www.facebook.com	www.youtube.com

Table 3: Excerpt from the list of manually selected dependable hosts (negative samples).

about 3,300,000 different sentences were acquired pertaining to $\approx$55,000 different context indicators.

Extracting Negative Examples

There are trustworthy sources of positive examples such as $Wikipedia$. However, there is no well-known authoritative source of diverse negative training material. For this, the manually tagged set of negative examples obtained in the previous section was extended as follows:

- A list of hosts was manually compiled, from which $\approx$52,500 negative samples were extracted (see table 3). The top forty highest frequent elements were picked, and the remaining sentences originated from these forty websites were selected as candidate negative samples.
- These negative candidates were scored according to a uni-gram LM deduced from the same set of $\approx$52,500 manually labeled instances. Here, an empirical threshold (0.0013) acted as a referee between negative and unlabelled instances. In this LM, stop-words and context indicators were left unconsidered. This assisted us in expanding the negative set to 118,871 instances with relation to 9,453 definienda.

Unlike other strategies, our corpus construction approach is neither fully automatic (Androutsopoulos and Galanis 2005) nor entirely manual (Xu et al. 2005; Fahmi and Bouma 2006). It also differs in that it is partially based on a collection of "*authoritative*" non-descriptive web-sites.

Building Balanced Training Sets

A major advantage of context models is that they can separate positive instances into several smaller groups (contexts). Accordingly, in order to improve performance, this work hypothesizes that there is no critical need for a massive collection of negative sentences, but rather only a set of about the same size as the larger positive context.

Firstly, a *default context* is generated by removing all cluster models that contain less than fifteen positive instances, so that data-sparseness is controlled. This reduced the ca. 55,000 contexts to 9,962 including the default category. At testing time, any unseen instance mismatching all context models is rated in agreement with this default class. The threshold of fifteen samples was picked so as to keep it as close as possible to the maximum amount of available negative samples. In practice, this implies 125,661 positive and all 118,871 negative sentences.

Secondly, balanced training sets were then constructed as follows. In each context, the same number of negative examples was selected in the next order: first from sentences matching and later mismatching the respective context indicator. In the former group, manually labeled are preferred to automatically tagged. In the latter group, the order is given by a unigram context language model inferred from the positive set so that samples with lexical ambiguity are preferred. Sentences are picked until the negative class has the same number of instances than the positive class. Note that automatically annotated examples were always required, in practice.

Extracting Testing Instances

Unlike training instances, testing samples were harvested from full-documents so that the impact of those samples extracted from truncated web-snippets can be taken into account. Our evaluation uses a set of 2,319 definienda randomly extracted from $FreeBase$, and which at the same time, were not present in $Wikipedia$. Documents were extracted from the Web by sending the definiendum in quotes to the $Bing$ search engine so as to retrieve the web-pages. In order to speed up this phase, only hits bearing the exact match of the definiendum within the web snippet were downloaded. Documents were then pre-processed similarly to entries from $Wikipedia$, and sentences observing the definition patterns in table 1 were chosen.

Finally, these instances were then manually labeled, and segmented into two balanced sets. Note that this annotation process was performed before any substitution of an entity with a placeholder. As a result, a set of unseen samples was obtained which is composed of 6,644 (6,250 different) instances containing 203 definienda, whereas a development set contained 6,630 examples belonging to 2,114 definienda. All testing definienda are linked with more than ten test samples, so to make sure that *precision at* k ($P@k$) can be computed for at least $k = 10$ in each case. Note that our model is only focused on ranking candidate answers and so, $Recall$ and therefore F-score are not suitable metrics as they assess the impact of other tasks such as redundancy removal which are independent QA components (Voorhees 2003). Hence we used $P@k$ which is a common measure for ranking answers.

Overall, the testing set involved 2,042 distinct context models including the default, which is applied to 481 different context indicators. The development set was used to find features, parameters and empirical thresholds required by the baselines.

Features Employed in our Models

For experiment purposes, different features were tested which involved diverse semantic and syntactic properties at both the surface and linguistic levels. In this case, the **number of tokens** in the sentence was seen as an attribute (Xu et al. 2005). Another property comes from the idea of **selective substitutions** (Cui, Kan, and Chua 2004).

Like many definitional QA systems (Zhang et al. 2005), our models were also enriched with **unigrams**. As extra surface elements, **bigrams**, **trigrams** and **word norms** (cf. (Church and Hanks 1990)) were also incorporated into our models. Moreover, eight boolean properties are incorporated (Miliaraki and Androutsopoulos 2004; Androutsopoulos and Galanis 2005), representing the matching of each of the **eight definition patterns** in table 1. Our models were so provided with with **nine boolean attributes** that are indicative of the existence of nine distinct relationships between the context indicator and any of the words in their respective sentence. These relations are determined via $WordNet$ and involved antonyms, pertainyms, hypernyms, hyponyms, holonyms, etc.

The remaining attributes are extracted from the lexicalised dependency graph representation of the sentences (Figueroa 2010). In the first place, bigrams, trigrams and tetragrams were acquired by following the *gov→dep* connections ruled by the trees. These three attributes are referred to as **bigrams-dp**, **trigrams-dp**, and **tetragrams-dp**, respectively. Two attributes regard the path from the root node to the definiendum: **root-to-definiendum-path** and **root-to-definiendum-distance**. The former enriches our models with the actual sequence of *gov→dep* links, whereas the latter does it with an element that denotes the number of nodes existing in this path.

Our models also account for **children of the root node paths**, which puts attributes representing the path from the root to each of its children.

Experiments and Results

In order to assess the effectiveness of our Maximum Entropy Context (MEC) models[5], comparisons were carried out by using three main baselines: centroid vector models (CV), bi-terms language models (B-LM) and context language models (C-LM):

Baseline I (CV) discriminates candidate sentences based on their similarity to the centroid vector of the respective definiendum (Xu, Licuanan, and Weischedel 2003; Cui et al. 2004). Since implementations are slightly different between each other, the blueprint proposed in (Chen, Zhon, and Wang 2006) was used for its construction. A centroid vector was built for each definiendum from a maximum of 330 web snippets (i.e., samples) fetched from $Bing$ Search. As a search strategy, multiple queries per definiendum were submitted organised as follows:

1. One query containing the definiendum and three task specific cues: "*biography*", "*dictionary*" and "*encyclopedia*".
2. Ten additional queries conforming to the structures listed in table 1. Note that since the first pattern typically yields manifold hits, this was divided into three distinct queries.

Baseline II (B-LM) profits from the ca. 3,300,000 preprocessed positive training material. This baseline is based on the bi-term language models of (Chen, Zhon, and Wang 2006) inferred from these training sentences. Accordingly, the mixture weight of these models was experimentally set to 0.84 by using the *Expectation Maximization* algorithm (Dempster, Laird, and Rubin 1977), and its reference length was experimentally set to fifteen words. Overall, this baseline aimed at testing the performance of the bi-term LMs (Chen, Zhon, and Wang 2006), but built on our training sets, against our testing sentences.

Baseline III (C-LM) implements the original context language models, but these are built on top of our positive sets and compared against our testing instances. This baseline differs from the original technique in the default model and the use of secondary context indicators.

Feature Selection

Features were automatically combined by using a greedy selection algorithm (Surdeanu, Ciaramita, and Zaragoza 2008). It incrementally selects features without assuming any of them as fixed. At each iteration, the procedure tests all attributes individually, and separates the one with the high increment in average Precision at k between all definienda. Since each definiendum is related to a distinct number of testing cases, the number of positive samples was used as k.

Once a property is chosen and fixed, each of the remaining features is dealt with the fixed attributes. When no addition brings about an improvement, the selection procedure stops, and the set of fixed properties is returned as the best set. Note that one best set of features was determined globally instead of selecting it per context so that the data-sparseness of small contexts is mitigated.

The final list of best features, when employing our development set, included: bigrams-dp, definition patterns 2 and 6, root node to children paths, and root to definiendum path. However, bear in mind the following issues:

1. The **number of tokens** was not selected, which reveals that the difference between the length of the instances extracted from web-snippets and $Wikipedia$ was not relevant. This supports their use for extracting negative instances.
2. Information derived from top level sequences/linked words provided by the hierarchy of the dependency tree was shown to be the most salient indicator of descriptive content.
3. Properties such as **bigrams-dp** were preferred to conventional **bigrams** and **word norms**. This implies the positive contribution of NLP-tools, namely linguistic knowledge distilled from lexical dependency relations.
4. While other features (i.e., unigrams, selective substitutions) have been widely used by other QA systems (Cui, Kan, and Chua 2004; Chen, Zhon, and Wang 2006; Fahmi and Bouma 2006; Qiu et al. 2007), these were proved to be not that effective in our approach.

[5] We used the implementation of Maximum Entropy Models provided at http://maxent.sourceforge.net/howto.html.

System	P@1	P@2	P@3	P@4	P@5	P@6	P@7	P@8	P@9	P@10
Baseline I (CV)	34.65	36.58	36.50	36.51	36.67	36.70	37.00	31.93	33.50	33.42
Baseline II (B-LMs)	50.50	47.82	47.03	46.95	46.84	46.25	45.68	48.76	49.50	48.02
Baseline III (C-LMs)	64.18	61.94	61.19	60.70	60.93	61.19	61.13	**64.93**	**63.85**	**63.18**
MEC	**69.46**	**66.75**	**65.52**	**65.15**	**64.33**	**63.38**	**62.42**	60.71	59.66	57.19
	+8.23%	+7.77%	+7.08%	+7.33%	+5.58%	+3.58%	+2.11%	-6.50%	-6.56%	-9.48%

Table 4: Precision at k (k=1...10) for different baseline and the proposed approach.

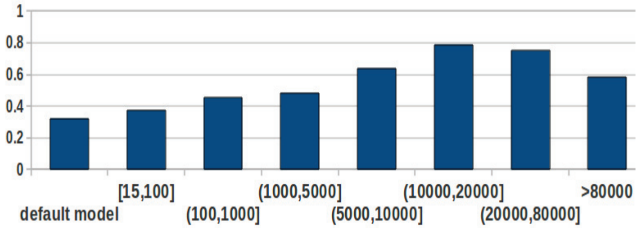

Figure 1: Proportion of Correct Predictions vs. Size of the Training Corpus

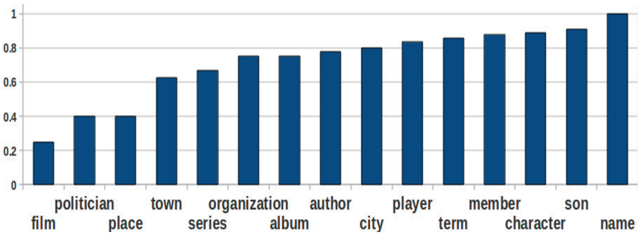

Figure 2: Proportion of Correct Predictions vs. Most Relevant Contexts in the Testing Set (Only larger contexts are described)

Model Assessment

The results of our experiments conducted for the previously described baselines compared with our MEC can be seen in table 4. The outcomes of the MEC improved the precision for the top seven places of the ranking, whereas it worsened for the last three positions. More precisely, this increased by more than 5% for the top five places, while for over 2% for the sixth and seventh positions. It shows that correctly classified candidate answers tend to concentrate more on top positions of the rank, while misclassified candidates are in the lower positions.

Experiments also showed that our negative corpus cooperates on increasing the performance. In other words, these data provided valuable information on regularities that characterise non-descriptions. More important, this improvement was achieved without ad-hoc negative data, namely specific samples for each context, but rather by the application of a subset of the same general-purpose/misc negative corpus to all contexts. This may indicate that better outcomes could be produced by generating specific training sets for each context (i.e., data matching the same context indicator).

The precision metrics also confirm that Bi-term language models perform better than centroid vector models (Chen, Zhon, and Wang 2006). This also stresses the fact that the extension of context LMs outperforms those Bi-terms LMs, and the strategy is outperformed by the extension proposed in the present work. Interestingly enough, the results reveal that Maximum Entropy Models are better approaches than LMs to deal with definition questions.

Figure 1 shows the results in terms of the size of contexts. The graph suggests that the larger the training set for building the models, the better the results are which is statistically significant based on the Pearson correlation coefficient (r = 0.52). However, the group involving the largest contexts seemed to get worse results than other large groups. This was due to the context containing sentences signalled with a named entity as indicator, thus solely 217 sentences out of 439 testing cases (49%) were correctly classified.

Incidentally, the graph also suggests that specific context models are better than general-purpose models as the performance of the default model was detrimental. Specifically, only 32% of the 481 answers scored by this model were correctly classified. This result also helps to understand the disappointing figures obtained by the context signalled by named entities as it also gathers samples from a wide semantic range.

Finally, figure 2 shows the performance of fifteen contexts containing a large training set. This also highlights the good performance achieved by these contexts. This also suggests that the merge of some contexts can boost the performance, more precisely, by combining large contexts with their semantically related small models (e.g., singer and/or composer with co-lyricist), this way the use of the default model can be reduced.

Conclusions

This work proposes Maximum Entropy Context models for ranking biographical answers to definition queries on the Internet. These answers are mined from web-documents, and different experiments show the promise of the approach to outperform other definitional state-of-the-art models.

Our model also found that web-snippets returned by commercial search engines are a fruitful source of negative examples, which can counteract positive instances extracted from KBs. In addition, the performance of MEC models is significantly correlated with the size of the training set acquired for each context. It suggests the merge of semantically connected contexts so as to reduce the impact of the data-sparseness of small contexts. Finally, experiments showed that features extracted from dependency trees can also be discriminative, especially *gov→dep* connections. A free implementation of the MEC models will be available to download at http://www.inf.udec.cl/∼atkinson/mecmodel.html.

Acknowledgments

This work is partially sponsored by the technology project **FONDEF** (Chile) under grant number **D08I1155:** *"Semantic Identification and Automatic Synthesis of Educational Materials for Specialised Domains"*

References

Androutsopoulos, I., and Galanis, D. 2005. A practically Unsupervised Learning Method to Identify Single-Snippet Answers to Definition Questions on the web. In *HLT/EMNLP*, 323–330.

Belkin, M., and Goldsmith, J. 2002. Using eigenvectors of the bigram graph to infer grammatical features and categories. In *Proceedings of the Morphology/Phonology Learning Workshop of ACL-02*.

Chen, Y.; Zhon, M.; and Wang, S. 2006. Reranking Answers for Definitional QA Using Language Modeling. In *Coling/ACL-2006*, 1081–1088.

Church, K. W., and Hanks, P. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics* 16(1):22–29.

Cui, H.; Li, K.; Sun, R.; Chua, T.-S.; and Kan, M.-Y. 2004. National University of Singapore at the TREC 13 Question Answering Main Task. In *Proceedings of TREC 2004*. NIST.

Cui, H.; Kan, M.-Y.; and Chua, T.-S. 2004. Unsupervised learning of soft patterns for definitional question answering. In *Proceedings of the Thirteenth World Wide Web Conference (WWW 2004)*, 90–99.

Cui, T.; Kan, M.; and Xiao, J. 2004. A Comparative Study on Sentence Retrieval for Definitional Question Answering. In *SIGIR Workshop on Information Retrieval for Question Answering (IR4QA)*, 383–390.

Dempster, A.; Laird, N.; and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society* 39:1–38.

Echihabi, A.; Hermjakob, U.; Hovy, E.; Marcu, D.; Melz, E.; and Ravichandran, D. 2003. Multiple-Engine Question Answering in TextMap. In *Proceedings of TREC 2003*, 772–781. NIST.

Fahmi, I., and Bouma, G. 2006. Learning to Identify Definitions using Syntactic Features. In *Proceedings of the Workshop on Learning Structured Information in Natural Language Applications*.

Figueroa, A., and Atkinson, J. 2009. Using Dependency Paths For Answering Definition Questions on The Web. In *WEBIST 2009*, 643–650.

Figueroa, A. 2008. Mining Wikipedia for Discovering Multilingual Definitions on the Web. In *Proceedings of the 4th International Conference on Semantics, Knowledge and Grid*.

Figueroa, A. 2010. *Finding Answers to Definition Questions on the Web*. Phd-thesis, Universitaet des Saarlandes.

Firth, J. R. 1957. A synopsis of linguistic theory 1930-1955. *In Studies in Linguistic Analysis* 1–32.

H. Joho and M. Sanderson. 2000. Retrieving Descriptive Phrases from Large Amounts of Free Text. In *9th ACM - CIKM*, 180–186.

H. Joho and M. Sanderson. 2001. Large Scale Testing of a Descriptive Phrase Finder. In *1st Human Language Technology Conference*, 219–221.

Han, K.; Song, Y.; and Rim, H. 2006. Probabilistic Model for Definitional Question Answering. In *Proceedings of SIGIR 2006*, 212–219.

Harris, Z. 1954. Distributional structure. In *Distributional structure. Word, 10(23)*, 146–162.

Hildebrandt, W.; Katz, B.; and Lin, J. 2004. Answering Definition Questions Using Multiple Knowledge Sources. In *Proceedings of the HTL-NAACL*, 49–56.

Katz, B.; Felshin, S.; Marton, G.; Mora, F.; Shen, Y. K.; Zaccak, G.; Ammar, A.; Eisner, E.; Turgut, A.; and Westrick, L. B. 2007. CSAIL at TREC 2007 Question Answering. In *Proceedings of TREC 2007*. NIST.

Lampouras, G., and Androutsopoulos, I. 2009. Finding Short Definitions of Terms on Web Pages. In *Proceedings of the 2009 EMNLP Conference*, 1270–1279.

Miliaraki, S., and Androutsopoulos, I. 2004. Learning to identify single-snippet answers to definition questions. In *COLING '04*, 1360–1366.

Qiu, X.; Li, B.; Shen, C.; Wu, L.; Huang, X.; and Zhou, Y. 2007. FDUQA on TREC2007 QA Track. In *Proceedings of TREC 2007*. NIST.

Rose, D. E., and Levinson, D. 2004. Understanding User Goals in Web Search. In *WWW*, 13–19.

Sacaleanu, B.; Neumann, G.; and Spurk, C. 2008. DFKI-LT at QA@CLEF 2008. In *In Working Notes for the CLEF 2008 Workshop*.

Surdeanu, M.; Ciaramita, M.; and Zaragoza, H. 2008. Learning to Rank Answers on Large Online QA Collections. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL 2008)*, 719–727.

Voorhees, E. M. 2003. Evaluating answers to definition questions. In *Proceedings of the HLT/NAACL*, 109–111.

Wu, L.; Huang, X.; Zhou, Y.; Zhang, Z.; and Lin, F. 2005. FDUQA on TREC2005 QA Track. In *Proceedings of TREC 2005*. NIST.

Xu, J.; Cao, Y.; Li, H.; and Zhao, M. 2005. Ranking Definitions with Supervised Learning Methods. In *WWW2005*, 811–819.

Xu, J.; Licuanan, A.; and Weischedel, R. 2003. TREC2003 QA at BBN: Answering Definitional Questions. In *Proceedings of TREC 2003*, 98–106. NIST.

Zhang, Z.; Zhou, Y.; Huang, X.; and Wu, L. 2005. Answering Definition Questions Using Web Knowledge Bases. In *Proceedings of IJCNLP 2005*, 498–506.