An Overview of Microsoft Web N-gram Corpus and Applications

Kuansan Wang **Christopher Thrasher** **Evelyne Viegas**
Xiaolong Li **Bo-june (Paul) Hsu**
Microsoft Research
One Microsoft Way
Redmond, WA, 98052, USA
webngram@microsoft.com

Abstract

This document describes the properties and some applications of the Microsoft Web N-gram corpus. The corpus is designed to have the following characteristics. First, in contrast to static data distribution of previous corpus releases, this N-gram corpus is made publicly available as an XML Web Service so that it can be updated as deemed necessary by the user community to include new words and phrases constantly being added to the Web. Secondly, the corpus makes available various sections of a Web document, specifically, the body, title, and anchor text, as separates models as text contents in these sections are found to possess significantly different statistical properties and therefore are treated as distinct languages from the language modeling point of view. The usages of the corpus are demonstrated here in two NLP tasks: phrase segmentation and word breaking.

1 Introduction

Since Banko and Brill's pioneering work almost a decade ago (Banko and Brill 2001), it has been widely observed that the effectiveness of statistical natural language processing (NLP) techniques is highly susceptible to the data size used to develop them. As empirical studies have repeatedly shown that simple algorithms can often outperform their more complicated counterparts in wide varieties of NLP applications with large datasets, many have come to believe that it is the size of data, not the sophistication of the algorithms that ultimately play the central role in modern NLP (Norvig, 2008). Towards this end, there have been considerable efforts in the NLP community to gather ever larger datasets, culminating the release of the English Giga-word corpus (Graff and Cieri, 2003) and the 1 Tera-word Google N-gram (Thorsten and Franz, 2006) created from arguably the largest text source available, the World Wide Web.

Recent research, however, suggests that studies on the document body alone may no longer be sufficient in understanding the language usages in our daily lives. A document, for example, is typically associated with multiple text streams. In addition to the document body that contains the bulk of the contents, there are also the title and the filename/URL the authors choose to name the document. On the web, a document is often linked with anchor text or short messages from social network applications that other authors use to summarize the document, and from the search logs we learn the text queries formulated by the general public to specify the document. A large scale studies reveal that these text streams have significantly different properties and lead to varying degrees of performance in many NLP applications (Wang *et al*, 2010, Huang *et al*, 2010). Consequently from the statistical modeling point of view, these streams are better regarded as composed in distinctive languages and treated as such.

This observation motivates the creation of Microsoft Web N-gram corpus in which the materials from the body, title and anchor text are made available separately. Another notable feature of the corpus is that Microsoft Web N-gram is available as a cross-platform XML Web service[1] that can be freely and readily accessible by users through the Internet anytime and anywhere. The service architecture also makes it straightforward to perform on

[1] Please visit http://research.microsoft.com/web-ngram for more information.

Proceedings of the NAACL HLT 2010: Demonstration Session, pages 45–48,
Los Angeles, California, June 2010.

demand updates of the corpus with the new contents that can facilitate the research on the dynamics of the Web.[2]

2 General Model Information

Like the Google N-gram, Microsoft Web N-gram corpus is based on the web documents indexed by a commercial web search engine in the EN-US market, which, in this case, is the Bing service from Microsoft. The URLs in this market visited by Bing are at the order of hundreds of billion, though the spam and other low quality web pages are actively excluded using Bing's proprietary algorithms. The various streams of the web documents are then downloaded, parsed and tokenized by Bing, in which process the text is lowercased with the punctuation marks removed. However, no stemming, spelling corrections or inflections are performed.

Unlike the Google N-gram release which contains raw N-gram counts, Microsoft Web N-gram provides open-vocabulary, smoothed back-off N-gram models for the three text streams using the CALM algorithm (Wang and Li, 2009) that dynamically adapts the N-gram models as web documents are crawled. The design of CALM ensures that new N-grams are incorporated into the models as soon as they are encountered in the crawling and become statistically significant. The models are therefore kept up-to-date with the web contents. CALM is also designed to make sure that duplicated contents will not have outsized impacts in biasing the N-gram statistics. This property is useful as Bing's crawler visits URLs in parallel and on the web many URLs are pointing to the same contents. Currently, the maximum order of the N-gram available is 5, and the numbers of N-grams are shown in Table 1.

Table 1: Numbers of N-grams for various streams

	Body	Title	Anchor
1-gram	1.2B	60M	150M
2-gram	11.7B	464M	1.1B
3-gram	60.1B	1.4B	3.2B
4-gram	148.5B	2.3B	5.1B
5-gram	237B	3.8B	8.9B

[2] The WSDL for the web service is located at http://web-ngram.research.microsoft.com/Lookup.svc/mex?wsdl.

CALM algorithm adapts the model from a seed model based on the June 30, 2009 snapshot of the Web with the algorithm described and implemented in the MSRLM toolkit (Nguyen et al, 2007). The numbers of tokens in the body, title, and anchor text in the snapshot are of the order of 1.4 trillion, 12.5 billion, and 357 billion, respectively.

3 Search Query Segmentation

In this demonstration, we implement a straightforward algorithm that generates hypotheses of the segment boundaries at all possible placements in a query and rank their likelihoods using the N-gram service. In other words, a query of T terms will have 2^{T-1} segmentation hypotheses. Using the famous query "mike siwek lawyer mi" described in (Levy, 2010) as an example, the likelihoods and the segmented queries for the top 5 hypotheses are shown in Figure 1.

Figure 1: Top 5 segmentation hypotheses under body, title, and anchor language models.

As can be seen, the distinctive styles of the languages used to compose the body, title, and the anchor text contribute to their respective models producing different outcomes on the segmentation

task, many of which research issues have been explored in (Huang *et al*, 2010). It is hopeful that the release of Microsoft Web N-gram service can enable the community in general to accelerate the research on this and related areas.

4 Word Breaking Demonstration

Word breaking is a challenging NLP task, yet the effectiveness of employing large amount of data to tackle word breaking problems has been demonstrated in (Norvig, 2008). To demonstrate the applicability of the web N-gram service for the work breaking problem, we implement the rudimentary algorithm described in (Norvig, 2008) and extend it to use body N-gram for ranking the hypotheses. In essence, the word breaking task can be regarded as a segmentation task at the character level where the segment boundaries are delimitated by white spaces. By using a larger N-gram model, the demo can successfully tackle the challenging word breaking examples as mentioned in (Norvig, 2008). Figure 2 shows the top 5 hypotheses of the simple algorithm. We note that the word breaking algorithm can fail to insert desired spaces into strings that are URL fragments and occurred in the document body frequently enough.

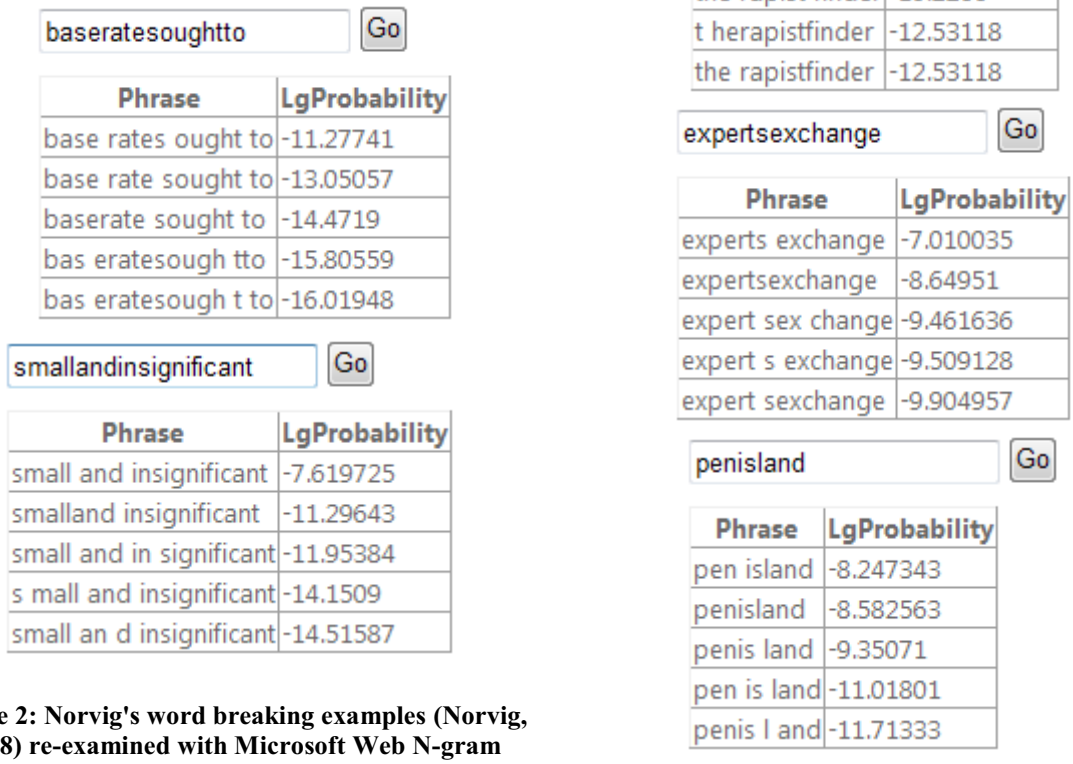

Figure 2: Norvig's word breaking examples (Norvig, 2008) re-examined with Microsoft Web N-gram

Two surprising side effects of creating the N-gram models from the web in general are worth noting. First, as more and more documents contain multi-lingual contents, the Microsoft Web N-gram corpus inevitably include languages other than EN-US, the intended language. Figure 3 shows examples in German, French and Chinese (Romanized) each.

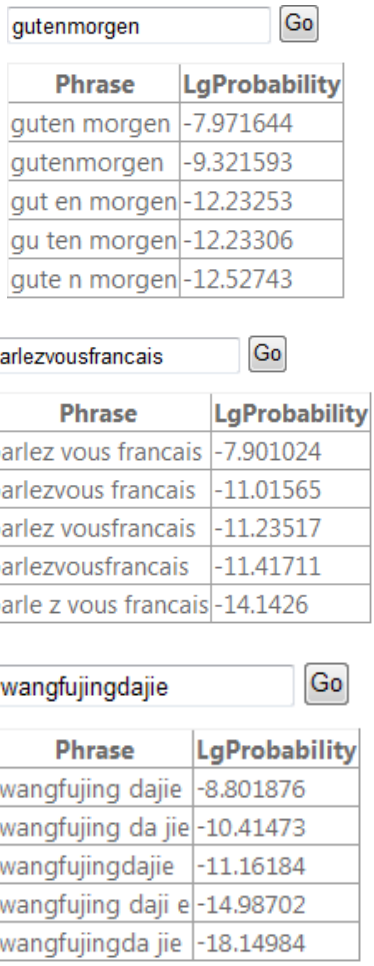

Figure 3: Word breaking examples for foreign languages: German (top), French and Romanized Chinese

Secondly, since the web documents contain many abbreviations that are popular in short messaging, the consequent N-gram model lends the simple word breaking algorithm to cope with the common short hands surprisingly well. An example that decodes the short hand for "wait for you" is shown in Figure 4.

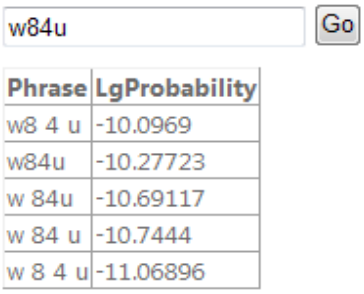

Figure 4: A word breaking example on SMS-style message.

References

Thorsten Brants and Alex Franz. 2006. *Web 1T 5-gram Version 1*. Linguistic Data Consortium, ISBN: 1-58563-397-6, Philadelphia.

Michel Banko and Eric Brill. 2001. Mitigating the paucity-of-data problem: exploring the effect of training corpus size on classifier performance for natural language processing. *Proc. 1st Internal Conference on human language technology research*, 1-5, San Diego, CA.

David Graff and Christopher Cieri. 2003. *English Gigaword*. Linguistic Data Consortium, ISBN: 1-58563-260-0, Philadelphia.

Jian Huang, Jianfeng Gao, Jiangbo Miao, Xiaolong Li, Kuansan Wang, and Fritz Behr. 2010. Exploring web scale language models for search query processing. In *Proc. 19th International World Wide Web Conference (WWW-2010),* Raleigh, NC.

Steven Levy, 2010. How Google's algorithm rules the web. *Wired Magazine*, February.

Patrick Nguyen, Jianfeng Gao, and Milind Mahajan. 2007. MSRLM: a scalable language modeling toolkit. *Microsoft Research Technical Report MSR-TR-2007-144*.

Peter Norvig. 2008. Statistical learning as the ultimate agile development tool. *ACM 17th Conference on Information and Knowledge Management Industry Event (CIKM-2008),* Napa Valley, CA.

Kuansan Wang, Jianfeng Gao, and Xiaolong Li. 2010. The multi-style language usages on the Web and their implications on information retrieval. In submission.

Kuansan Wang, Xiaolong Li and Jianfeng Gao, 2010. Multi-style language model for web scale information retrieval. In *Proc. ACM 33rd Conference on Research and Development in Information Retrieval (SIGIR-2010)*, Geneva, Switzerland.

Kuansan Wang and Xiaolong Li, 2009. Efficacy of a constantly adaptive language modeling technique for web scale application. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-2009)*, Taipei, Taiwan.