Predicting Words and Sentences using Statistical Models

Nicola Carmignani

Departement of Computer Science
University of Pisa

carmigna@di.unipi.it

Language and Intelligence Reading Group

July 5, 2006

Outline

1. Introduction

2. Word Prediction
   - The Origin of Word Prediction
   - n-gram Models

3. Sentence Prediction
   - Statistical Approach
   - Information Retrieval Approach

4. Conclusions

Outline

1. Introduction

2. Word Prediction
   - The Origin of Word Prediction
   - n-gram Models

3. Sentence Prediction
   - Statistical Approach
   - Information Retrieval Approach

4. Conclusions

Introduction

- Natural Language Processing (NLP) aims to study the problems of automated generation and understanding of natural human languages.

- The major tasks in NLP:
  - Text-to-Speech (TTS)
  - Speech Recognition
  - Machine Translation
  - Information Extraction
  - Question-Answering
  - Part-of-Speech (POS) Tagging
  - Information Retrieval
  - Automatic Summarization

Statistical NLP

- *Statistical Inference* aims to collect some data and then make some inferences about its probability distribution.
- Prediction issues require an appropriate language model. Natural language modeling is a statistical inference problem.
- Statisitcal NLP methods can be useful in order to capture “human” knowledge needed to allow prediction, and assess the likelihood of various hypotheses
  - probability of word sequences;
  - likelihood of words co-occurrence.

Prediction::An Overview

Humans are good in word prediction...

- *Once upon a*

... and sentence prediction.

- *Penny Lane*

Prediction::An Overview

Humans are good in word prediction...

- *Once upon a* time

... and sentence prediction.

- *Penny Lane*

Prediction::An Overview

Humans are good in word prediction...

- *Once upon a* time

... and sentence prediction.

- *Penny Lane*

Prediction::An Overview

Humans are good in word prediction...

- *Once upon a* time

... and sentence prediction.

- *Penny Lane* is in my ears and in my eyes

Prediction::Why?

- Predictors support writing and are commonly used in combination with assistive devices such as keyboards, virtual keyboards, touchpads and pointing devices.
- Frequently, applications include repetitive tasks such as writing emails in call centers or letters in an administrative environment.
- Applications of word prediction:
  - Spelling Checkers
  - Mobile Phone/PDA Texting
  - Disabled Users
  - Handwriting Recognition
  - Word-sense Disambiguation

Outline

1. Introduction
2. Word Prediction
    - The Origin of Word Prediction
    - n-gram Models
3. Sentence Prediction
    - Statistical Approach
    - Information Retrieval Approach
4. Conclusions

Word Prediction::An Overview

- *Word Prediction* is the problem of guessing which word is likely to continue a given initial text fragment.

- Word prediction techniques are well-established methods in the field of AAC (Augmentative and Alternative Communication) that are frequently used as communication aids for people with disabilities
  - accelerate the writing;
  - reduce the effort needed to type;
  - suggest the correct word (no misspellings).

Please, don't confuse!

- Usually, when I say "*word prediction*", everybody calls Tegic T9 to mind.
- T9 is a successful system but its prediction is based on dictionary disambiguation (only according to last word).
- We would like something that is skilful at doing prediction according to the previous context.

Word Prediction::The Origins

The word prediction task can be framed viewed as the statistical formulation of the speech recognition problem.

- Finding the most likely word sequence $\hat{W}$ given the observable acoustic signal
$$\hat{W} = \arg\max_{W} \mathbb{P}(W|A)$$
- We can rewrite it using Bayes' rule
$$\hat{W} = \arg\max_{W} \frac{\mathbb{P}(A|W)\mathbb{P}(W)}{\mathbb{P}(A)}$$
- Since $\mathbb{P}(A)$ is independent of the choice of W, we can simplify as follows
$$\hat{W} = \arg\max_{W} \mathbb{P}(A|W)\mathbb{P}(W)$$

n-gram Models::Introduction

- In order to predict the next word (w_N) given the context or history ($w_1, \ldots, w_{N-1}$), we want to estimate this probability function:

$$\mathbb{P}(w_N | w_1, \ldots, w_{N-1})$$

- The language model estimates the values $\mathbb{P}(W)$, where $W = w_1, \ldots, w_N$.

- By using Bayes theorem, we get

$$\mathbb{P}(W) = \prod_{i=1}^{N} \mathbb{P}(w_i | w_1, w_2 \ldots, w_{i-1})$$

n-gram Models

- Since the parameter space of $\mathbb{P}(w_i | w_1, w_2 \ldots, w_{i-1})$ is too large, we need a model where all similar histories $w_1, w_2 \ldots, w_{i-1}$ are placed in the same equivalence class.

- Markov Assumption: only the prior local content (the last few words) affects the next word.

$(n-1)^{th}$ **Markov Model** or **n-gram**

n-gram Models

Formally, n-gram model is denoted by:

$$\mathbb{P}(w_i | w_1, \ldots, w_{i-1}) \approx \mathbb{P}(w_i | w_{i-n+1}, \ldots, w_{i-1})$$

- Typical values of n-gram are
  - $n = 1$ (unigram)
    $\mathbb{P}(w_i \mid w_1, \ldots, w_{i-1}) \approx \mathbb{P}(w_i)$
  - $n = 2$ (bigram)
    $\mathbb{P}(w_i \mid w_1, \ldots, w_{i-1}) \approx \mathbb{P}(w_i \mid w_{i-1})$
  - $n = 3$ (trigram)
    $\mathbb{P}(w_i \mid w_1, \ldots, w_{i-1}) \approx \mathbb{P}(w_i \mid w_{i-2}\ w_{i-1})$

n-gram word Models::Example

Example:

- W = *Last night I went to the concert*
- Instead of $\mathbb{P}(\textit{concert} \mid \textit{Last night I went to the})$
- we use a bigram $\mathbb{P}(\textit{concert} \mid \textit{the})$
- or a trigram $\mathbb{P}(\textit{concert} \mid \textit{to the})$

How to Estimate Probabilities

- Where do we find these probabilities?
  - Corpora are collections of text and speech (e.g. Brown Corpus).
- Two different coprora are needed:
  - Probabilities are extracted from a training corpus, which is necessary to design the model.
  - A test corpus is used to run trials in order to evaluate the model.

Problems with n-grams

- The drawback of these methods is the amount of text needed to train the model. Training corpus has to be large enough to ensure that each valid word sequence appears a relevant number of times.
- A great amount of computational resources is needed especially if the number of words in the lexicon is big.

For a vocabulary $\mathcal{V}$ of 20,000 words

- $|\mathcal{V}|^2 = 400$ million of bigrams;
- $|\mathcal{V}|^3 = 8$ trillion of trigrams;
- $|\mathcal{V}|^4 = 1.6 \times 10^{17}$ of four-grams.

- Since the number of possible words is very large, there is a need to focus attention on a smaller subset of these.

n-gram **POS** Models

- One proposed solution consists in generalizing the n-gram model, by grouping the words in *category* according to the context.
- A mapping φ is defined to approximate a context by means of the equivalence class it belongs to: $\mathbb{P}(w_i|\varphi[w_{i-n+1}, \ldots, w_{i-1}])$.
- Usually, Part-of-Speech (POS) tags are used as mapping function, replacing each word with the corresponding POS tag (i.e. classification).
- POS tags have the potential of allowing generalization over similar words, as well as reducing the size of the language model.

n-gram Models for Inflected Languages

- Many word prediction methods are focused on non-inflected languages (English) that have a small amount of variation.

- Inflected languages can have a huge amount of affixes that affect the syntactic function of every word. It is difficult to include every variation of a word in the dictionary.

- Italian is a very morphologically rich language with a high rate of inflected forms. A morpho-syntactic component is needed to compose inflections in accordance with the context
  - Gender: "lui è un" ..."professore" not "professoressa";
  - Number: "le mie" ... "scarpe" not "scarpa";
  - Verbal agreement: "la ragazza" ... "scrive" not "scriviamo".

Hybrid Approach to Prediction

- Prediction can either be based on text statistics or linguistic rules.

- Two Markov models can be included: one for word classes (POS tag unigrams, bigrams and trigrams) and one for words (word unigrams and bigrams). A linear combination algorithm may combine these two models.

- Incorporating morpho-syntactic information to enforce prediction accuracy.

Outline

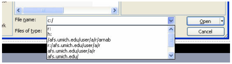
1. Introduction
2. Word Prediction
   - The Origin of Word Prediction
   - n-gram Models
3. Sentence Prediction
   - Statistical Approach
   - Information Retrieval Approach
4. Conclusions

Sentence Prediction::An Overview

- It's now easy to presume what is Sentence Prediction. Now we would like to predict how a user will continue a given initial fragment of natural language text.

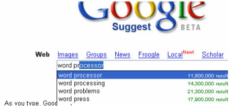
- Some Applications:
  - Korvemaker and Greiner have developed a system which predicts whole command lines.

OS



File name: c:/

r:
h:
/afs.umich.edu/user/a/r/arnab
r:/afs.umich.edu/user/a/r
afs.umich.edu/user/a/r
afs.umich.edu/



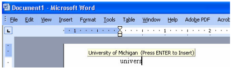
word processor

word processor	11,800,000 result
word processing	14,300,000 result
word problems	21,300,000 result
word press	17,900,000 result

As you type, Good

Search Engine

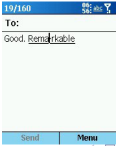
Word Processor



Document1 - Microsoft Word

University of Michigan (Press ENTER to Insert)

univers



19/160

To:

Good. Remarkable

Mobile Phone

Images retrieved from Arnab Nandi presentation: *Better Phrase Prediction*

Sentence Prediction::Two Approaches

- A possible approach for sentence prediction problem might be to learn a language model and to construct the most likely sentence: statistical approach.

- An alternative solution to address completion might involve information retrieval methods.

- Domain specific collection of documents are used in both researces as corpora. Clearly, a constrained application context improves the accuracy of prediction.

Sentence Prediction::Statistical Approach

- As shown, n-gram language models provide a natual approach to the construction of sentence completion systems, but they could not be sufficient.

- Eng and Eisner have developed a radiology report entry system that implements an automated phrase completion feature based on language modeling (trigram language model).

- Bickel, Haider and Scheffer have developed an n-gram based completion method using specific document collections, such as emails and weather reports.

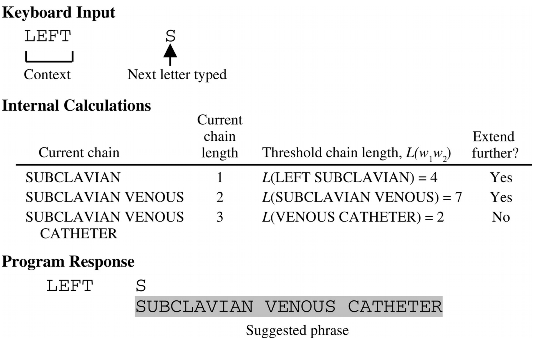
[Eng et al., 2004]

- Radiology report domain
  - Training corpus: 36,843 general reports.
  - Performance tested on 200 reports outside of the training set.

- The algorithm is based on a trigram language model and provides both word prediction and phrase completion.

- Word chaining guesses zero or more subsequent words.
  - A *threshold chain length* $L(w_1, w_2)$ can be determined in order to extend prediction to furher words.

[Eng et al., 2004]



Keyboard Input

LEFT S

Context

Next letter typed

Internal Calculations

Current chain	Current chain length	Threshold chain length, $L(w_1 w_2)$	Extend further?
SUBCLAVIAN	1	*L*(LEFT SUBCLAVIAN) = 4	Yes
SUBCLAVIAN VENOUS	2	*L*(SUBCLAVIAN VENOUS) = 7	Yes
SUBCLAVIAN VENOUS CATHETER	3	*L*(VENOUS CATHETER) = 2	No

Program Response

LEFT S

SUBCLAVIAN VENOUS CATHETER

Suggested phrase

- All alphabetic characters were converted to uppercase.
- Words occurring fewer than 10 times in the corpus were replaced with a special label in order to eliminate misspelled words.
- Punctuation marks were removed from the corpus, so they do not appear in the suggested sentence and must be entered when needed.

[Bickel et al., 2005]

- Application corpora: Call-Center emails, personal emails, weather reports and cooking recipes.

- The sentence completion is based on a linear interpolation of n-gram models
  - Finding the most likely word sequence $w_{t+1}, \ldots, w_{t+T}$ given a word n-gram model and an initial sequence $w_1, \ldots, w_t$.
  - The decoding problem is mathematically defined as follows: $\mathbb{P}(w_{t+1}, \ldots, w_{t+T} \mid w_1, \ldots, w_t)$
  - The n^{th} order Markov assumption constrains each w_t to be dependent on at most w_{t-n+1} through w_{t-1}.
  - The parameters of the problem are: $\mathbb{P}(w_t \mid w_{t-n+1}, \ldots, w_{t-1})$

[Bickel et al., 2005]

- An n-gram model is learned by estimating the probability of all possible combinations of n words.

- The solution to overcome data sparseness problem is to use a weighted linear mixture of n-gram models.

- Several mathematical transformations lead the problem to a Viterbi algorithm that retrieves the most likely word sequence.
  - This algorithm starts with the most recently entered word (w_t) and moves iteratively looking for highest scored periods.

Sentence Prediction::IR Approach

- An information retrieval approach to sentence prediction involves finding, in a corpus, the sentence which is most similar to a given initial fragment.
- Grabski and Scheffer have developed an indexing method that retrieves the sentence from a collection of documents.
- *Information retrieval* aims to provide methods that satisfy a user's information needs. Here, the model has to retrieve the remaining part of a sentence.

[Grabski et al., 2004]

- Research approach is to search for the sentence whose initial words are most similar to the given initial sequence in vector space representation.

- For each training sentence d_j and each length ℓ, a TF-IDF representation of the first ℓ words is calculated: $f_{i,j}^{\ell} = \mathit{normalize}(\mathit{TF}(t_i, d_j, \ell) \times \mathit{IDF}(t_i))$

- The similarity between two vectors is defined by the cosine measure.

[Grabski et al., 2004]

- To find the best fitting sentence an indexing algorithm is used
  - An inverse index structure lists, for each term, the sentences in which the term occurs (the postings).
  - The postings lists are sorted according to a relation “$<$” that is defined on sentence pairs: $s_1 < s_2$ if s_1 appears in the document collection more frequently than s_2.
  - A similarity bound can be calculated to stop the retrieval algorithm, because there is no better sentence left to find.

[Grabski et al., 2004]

- Such a structure improves access time but raises the problem of having to store a huge amount of data.

- Data compression has been achieved by using clustering techniques finding groups of semantically equivalent sentences.

- The result of clustering algorithm is a tree of clusters. The leaf nodes contain the groups of sentences. The tree can also be used to access the data more quickly.

Outline

1. Introduction

2. Word Prediction
   - The Origin of Word Prediction
   - n-gram Models

3. Sentence Prediction
   - Statistical Approach
   - Information Retrieval Approach

4. Conclusions

Conclusions

- A prediction system is particularly useful to minimize keystrokes for users with special needs and to reduce misspellings and typographic errors. Moreover, it can be effectively used in language learning, by suggesting well-formed words to non-native users.

- Prediction methods can include different modeling strategies for linguistic information.

- Stochastic modeling (n-gram models) considers a small amount of information of written text (e.g. the last n words).

It’s Worth Another Try!!!

THE

It's Worth Another Try!!!

THE END

References I

S. Hunnicutt, L. Nozadze and G. Chikoidze,
Russian Word Prediction with Morphological Support,
5th International Symposium on Language, Logic and Computation, Tbilisi, Georgia, 2003.

Y. Even-Zohar and D. Roth,
A Classification Approach to Word Prediction,
NAACL-2000, The 1st North American Conference on Computational Linguistics, 124–131, 2000.

S. Bickel, P. Haider and T. Scheffer,
Predicting Sentences using N-Gram Language Models,
Proceedings of Conference on Empirical Methods in Natural Language Processing, 2005.

References II

- A. Fazly and G. Hirst,
  Testing the Efficacy of Part-of-Speech Information in Word Completion,
  Proceedings of the Workshop on Language Modeling for Text Entry Methods, 10th EACL, Budapest, 2003.

- J. Eng and J. Eisner,
  Radiology Report Entry with Automatic Phrase Completion Driven by Language Modeling,
  Radiographics 24(5):1493–1501, September-October, 2004.

- K. Grabski and T. Scheffer,
  Sentence Completion,
  Proceedings of the SIGIR International Conference on Information Retrieval, 2004.

References III

B. Korvemaker and R. Greiner,
Predicting UNIX Command Lines: Adjusting to User Patterns,
Proceedings of AAAI/IAAI 2000: 230–235, 2000.

Cagigas S.,
Contribution to Word Prediction in Spanish and its Integration in Technical Aids for People with Physical Disabilities,
PhD Dissertation, Madrid University, 2001.

Gustavii E. and Pettersson E.,
A Swedish Grammar for Word Prediction,
Master's Thesis, Department of Linguistics at Uppsala University, 2003.