机器翻译

数据集 格式:

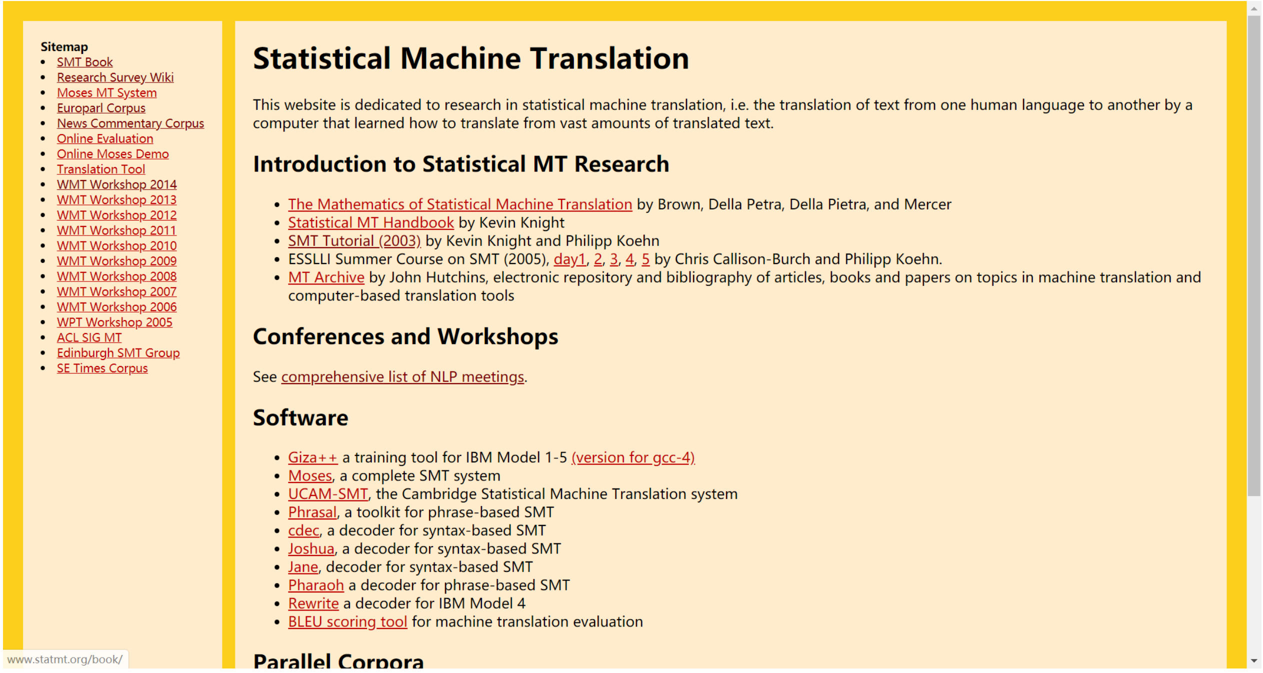
句子pairs:

$$\begin{array}{cc} english\ sentence_1 & chinese\ sentence_1 \\ english\ sentence_2 & chinese\ sentence_2 \\ english\ sentence_3 & chinese\ sentence_3 \\ & \cdots\cdots \\ english\ sentence_n & chinese\ sentence_n \end{array}$$

例子:

规模	源句子	目标句子	提供者	详情	备注
小规模	英语	越南语	IWSLT	133k sentence pairs	-
大规模	德语	英语	WMT	4.5M sentence pairs	-

WMT网站



Sitemap
- SMT Book
- Research Survey Wiki
- Moses MT System
- Europarl Corpus
- News Commentary Corpus
- Online Evaluation
- Online Moses Demo
- Translation Tool
- WMT Workshop 2014
- WMT Workshop 2013
- WMT Workshop 2012
- WMT Workshop 2011
- WMT Workshop 2010
- WMT Workshop 2009
- WMT Workshop 2008
- WMT Workshop 2007
- WMT Workshop 2006
- WPT Workshop 2005
- ACL SIG MT
- Edinburgh SMT Group
- SE Times Corpus

Statistical Machine Translation

This website is dedicated to research in statistical machine translation, i.e. the translation of text from one human language to another by a computer that learned how to translate from vast amounts of translated text.

Introduction to Statistical MT Research

- The Mathematics of Statistical Machine Translation by Brown, Della Petra, Della Pietra, and Mercer
- Statistical MT Handbook by Kevin Knight
- SMT Tutorial (2003) by Kevin Knight and Philipp Koehn
- ESSLLI Summer Course on SMT (2005), day1, 2, 3, 4, 5 by Chris Callison-Burch and Philipp Koehn.
- MT Archive by John Hutchins, electronic repository and bibliography of articles, books and papers on topics in machine translation and computer-based translation tools

Conferences and Workshops

See comprehensive list of NLP meetings.

Software

- Giza++ a training tool for IBM Model 1-5 (version for gcc-4)
- Moses, a complete SMT system
- UCAM-SMT, the Cambridge Statistical Machine Translation system
- Phrasal, a toolkit for phrase-based SMT
- cdec, a decoder for syntax-based SMT
- Joshua, a decoder for syntax-based SMT
- Jane, decoder for syntax-based SMT
- Pharaoh a decoder for phrase-based SMT
- Rewrite a decoder for IBM Model 4
- BLEU scoring tool for machine translation evaluation

Parallel Corpora

www.statmt.org/book/

数据集部分

ACL 2014 NINTH WORKSHOP
ON STATISTICAL MACHINE TRANSLATION

Shared Task: Machine Translation

26-27 June 2014
Baltimore, USA

[HOME] | [TRANSLATION TASK] | [METRICS TASK] | [QUALITY ESTIMATION TASK] | [MEDICAL TRANSLATION TASK] | [SCHEDULE] | [PAPERS] | [AUTHORS] | [RESULTS]

The recurring translation task of the WMT workshops focuses mainly on European language pairs, but this year we have introduced English-Hindi as an experimental, low resource language pair. Translation quality will be evaluated on a shared, unseen test set of news stories. We provide a parallel corpus as training data, a baseline system, and additional resources for download. Participants may augment the baseline system or use their own system.

GOALS

The goals of the shared translation task are:

- To investigate the applicability of current MT techniques when translating into languages other than English
- To examine special challenges in translating between European languages, including word order differences and morphology
- To investigate the translation of low-resource, morphologically rich languages
- To create publicly available corpora for machine translation and machine translation evaluation
- To generate up-to-date performance numbers for European languages in order to provide a basis of comparison in future research
- To offer newcomers a smooth start with hands-on experience in state-of-the-art statistical machine translation methods

We hope that both beginners and established research groups will participate in this task.

TASK DESCRIPTION

We provide training data for five language pairs, and a common framework (including a baseline system). The task is to improve methods current methods. This can be done in many ways. For instance participants could try to:

- improve word alignment quality, phrase extraction, phrase scoring
- add new components to the open source software of the baseline system
- augment the system otherwise (e.g. by preprocessing, reranking, etc.)

DOWNLOAD

- Parallel data:

File	Size	CS-EN	DE-EN	HI-EN	FR-EN	RU-EN	Notes
Europarl v7	628MB	✓	✓		✓		same as previous year, corpus home page
Common Crawl corpus	876MB	✓	✓		✓	✓	same as previous year
UN corpus	2.3GB				✓		same as previous year, corpus home page
News Commentary	77MB	✓	✓		✓	✓	updated, data with document boundaries
10^9 French-English corpus	2.3 GB				✓		same as previous year [md5|sha1]
CzEng 1.0	115MB	✓					same as previous year, corpus home page (avoid sections 98 and 99)
Yandex 1M corpus	121MB					✓	corpus home page; v1.3 now in original case
Wiki Headlines	7.8MB			✓		✓	Provided by CMU. The ru-en is unchanged from last year.
HindEnCorp	25MB			✓			Collected by Charles University
~~The JHU Corpus~~				~~✓~~			This is fully contained in HindEnCorp, so not made available here.

翻译系统

统计机器翻译

- **方法：**

将原句子分成短语块，查词典翻译

$$P(en \mid ch) = \frac{P(ch \mid en) * p(en)}{P(ch)}$$

P(ch)可以视为常数，写成：

$$P(en \mid ch) = argmax_{en} \quad P(ch \mid en) * P(en)$$

例子：

假如我们有语料库：

```
今天 是 个 好 天气。        Today is a fine day.
你 真是 太 可爱 了。        You are so cute.
你 今天 上午 有 课。        You have classes this morning.
接下来 有 很多 论文 要 读。  There are many papers to read next.
```

```
他 是个 爱 读书 的 孩子。    He is a child who likes reading.
是 时候 去 峡谷 乘凉 了。    It's time to cool down in the canyon.
亚瑟 就是 峡谷 中 的 王者。  Arthur is the king in the canyon.
这篇 论文 讲了 很多 模型，  This paper talks about a lot of models,
这些 模型 的 效果 很不错。  and the results of these models are very good.
```

翻译：我在 峡谷 中 读 论文。

$$
\begin{aligned}
P(en \mid ch) &= P(en \mid \text{“我 在 峡谷 中 读 论文”}) \\
&= P(\text{“我 在 峡谷 中 读 论文”} \mid w_1, w_2, w_3, w_4, w_5, w_6) * P(w_1, w_2, w_3, w_4, w_5, w_6) \\
&= A * B
\end{aligned}
$$

其中，可认为A是翻译模型，B是语言模型， B可以用来评价生成句子的质量：

$$
\begin{aligned}
&A = P(\text{“我”} \mid w_1)P(\text{“在”} \mid w_2)P(\text{“峡谷”} \mid w_3)P(\text{“中”} \mid w_4)P(\text{“读”} \mid w_5)P(\text{“论文”} \mid w_6) \\
&B = P(w_1, w_2, w_3, w_4, w_5, w_6) \\
&1) = P(w_1) * P(w_2) * P(w_3) * P(w_4) * P(w_5) * P(w_6) \qquad (unigram\text{形式}) \\
&2) = P(w_1) * P(w_2 \mid w_1) * P(w_3 \mid w_2) * P(w_4 \mid w_3) * P(w_5 \mid w_4) * P(w_6 \mid w_5) \qquad (bigram\text{形式}) \\
&3) = P(w_1) * P(w_2 \mid w_1) * P(w_3 \mid w_1, w_2) * P(w_4 \mid w_2, w_3) * P(w_5 \mid w_3, w_4) * P(w_6 \mid w_4, w_5) \qquad (trigram)
\end{aligned}
$$

B如何评价一个句子像不像人说的呢？我们为您B越大说明句子越像人话，事实上也是这样的，你可以计算下。1）形式的除外；将B写成对数和形式可以得到Perplex指标，越小句子越通顺。

- **缺点：**
  - 不流畅，无法利用语义信息；
  - 需要领域知识；
  - 计算耗时；

神经机器翻译(NMT)

1. 文本表示

我们需要将文本表示成数值(scalar/vector)形式，才能为给神经网络模型。历史上有很多将单词表示成向量的方法，他们各有优劣。

1.1 One-Hot Vector

假如我们的语料库只有一句话：

```
我 爱 北京 天安门 ， 他 也 爱。
```

则我们的词典有 $[\text{我，爱，北京，天安门，他，也，} <com>, <eos>]$ 8个单词，则上面句子可以表示成：

$$
\begin{aligned}
\text{"我"} \quad & [1,0,0,0,0,0,0,0] \\
\text{"爱"} \quad & [0,1,0,0,0,0,0,0] \\
\text{"北京"} \quad & [0,0,1,0,0,0,0,0] \\
\text{"天安门"} \quad & [0,0,0,1,0,0,0,0] \\
\text{"}<com>\text{"} \quad & [0,0,0,0,0,0,1,0] \\
\text{"他"} \quad & [0,0,0,0,1,0,0,0] \\
\text{"也"} \quad & [0,0,0,0,0,1,0,0] \\
\text{"爱"} \quad & [0,1,0,0,0,0,0,0] \\
\text{"}<eos>\text{"} \quad & [0,0,0,0,0,0,0,1]
\end{aligned}
$$

其中，每个词向量的维度都是 |V| (词典大小)

- 缺点

- 矩阵稀疏，维度大
- 无法表示单词的语义信息
- 没有使用上下文信息

1.2 BOW(bag of words)

一种基于词频的向量，假如我们有 $科技类docs1, 体育类docs2, 时政类docs3, 娱乐类docs4$，我们统计每类文档下面每个单词出现的次数：

words	docs1	docs2	docs3
我	10	3	30
汽车	50	10	2
...	...	...	...
中国	30	20	60

我们可以把每行拿出来代表这个单词的向量。

- 优点
  - 可以计算单词之间的相似性
  - 维度可以调整
- 缺点
  - 每个单词的词频有相同的重要性
  - 无法表示单词与单词之间的语义信息(上下文)
- 解决方法
  - tf-idf向量（解决每个文档中每个单词的重要性问题）（略）
  - word vector(利用上下文信息)

1.3 Word Embedding

1.3.1 使用语言模型（LM）训练

可以把语言模型裂解为评价一个句子像不像人说的话的模型；

$P(w_1, w_2, ..., w_n)$， 值越大越像人说的话。

1)神经网络语言模型(NNLM)

由前t-1个词预测下面一个单词(2003)：

$P(W_t = "some\_word" \mid W_1, W_2, ..., W_{t-1})$

2)Word2Vec

- CBOW

- 上下文预测中心词

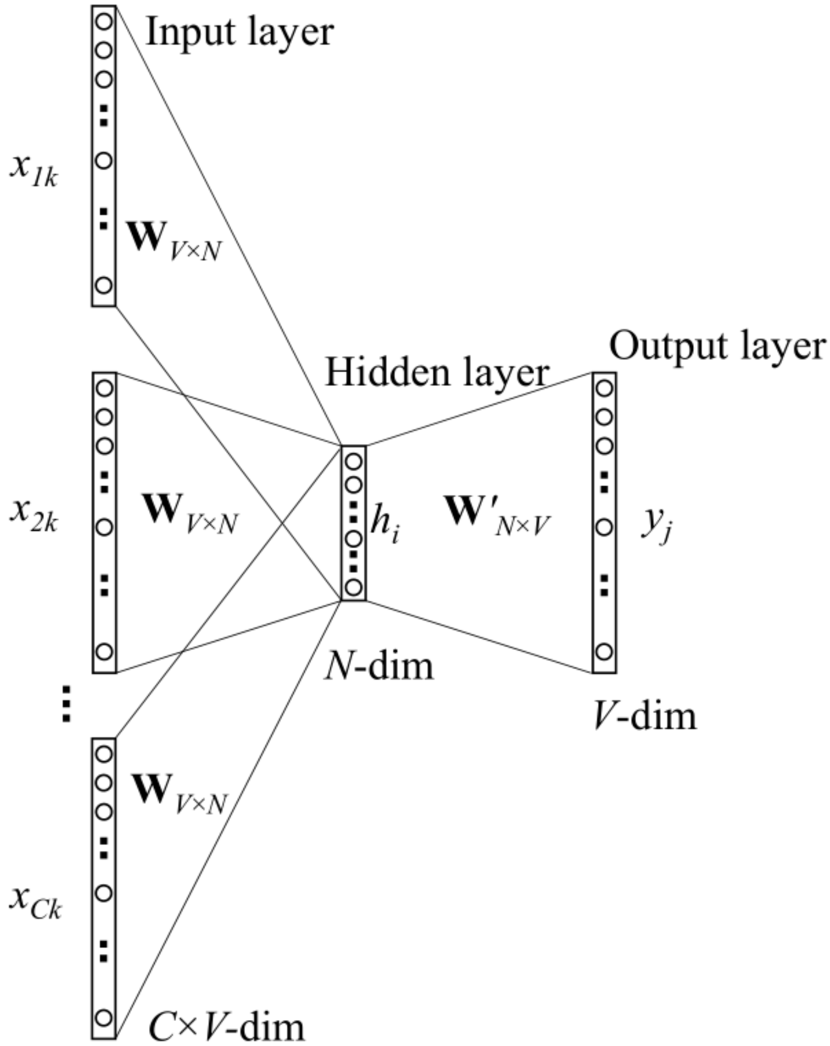


Figure 2: Continuous bag-of-word model

- Skip-gram

- 中心词预测上下文

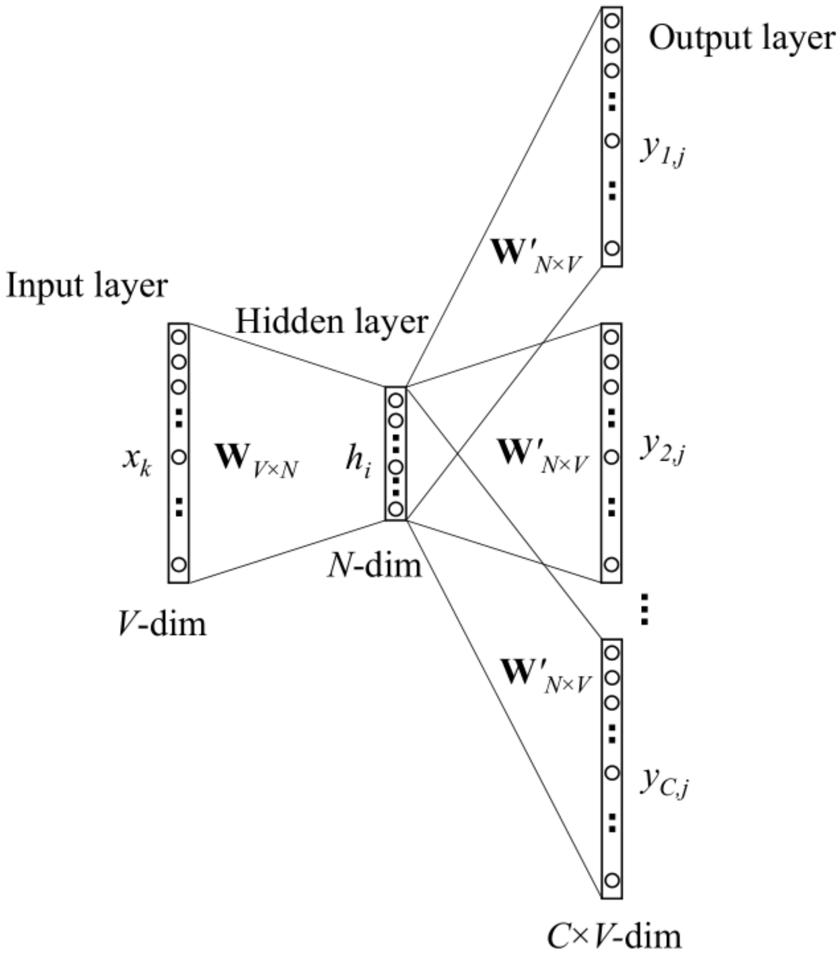


Figure 3: The skip-gram model.

- 优点
  - 利用单词之间的关系表示语义信息
  - 维度可控
- 缺点:
  - 多语义单词较难处理
- 解决方法 ELMo（RNN）, Bert（transformer）等预训练模型（略）

```
3）（Embedding from Language Models）ELMo

多层双向RNN训练

缺点：
     - RNN训练速度慢
     - 长时依赖信息很难提取

4）GPT

- transformer的decoder

5）Bert

- transformer的encoder
```

2. 翻译模型

神经机器翻译的整体框架目前主要是seq2seq模型，解决的是变长输入和变长输出的问题。

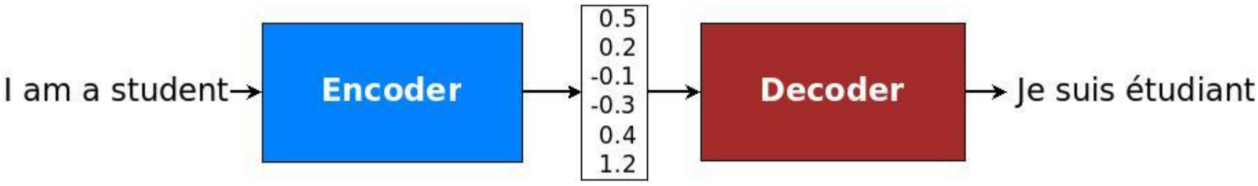
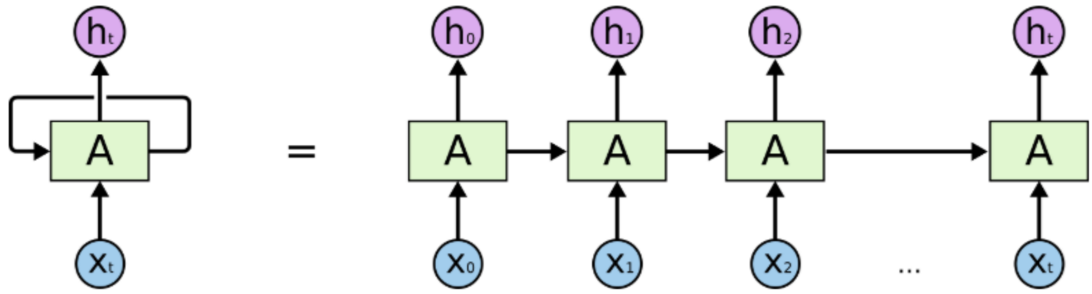


Figure 1. Encoder-decoder architecture *(源自： https://github.com/tensorflow/nmt#introduction)*

2.1 基于RNN的

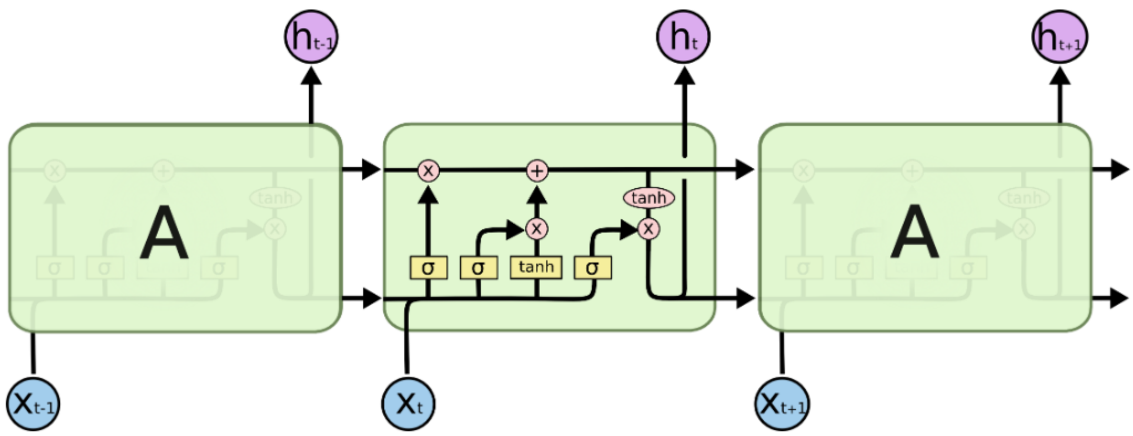
- 方法：通过RNN-encoder将源句子转化为一个vector(常成为这句话的contex vector)，再将该向量送入Decoder产生目标句子。

普通的RNN：



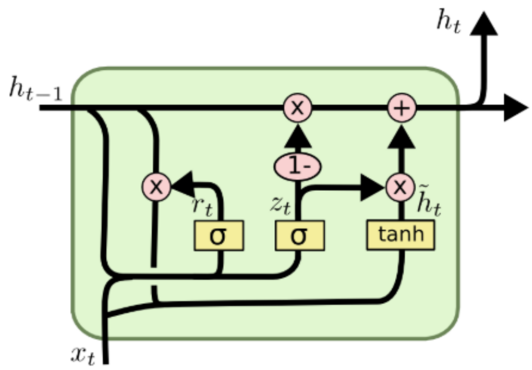
An unrolled recurrent neural network.

LSTM：



The repeating module in an LSTM contains four interacting layers.

GRU：



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$
$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

- 整体框架

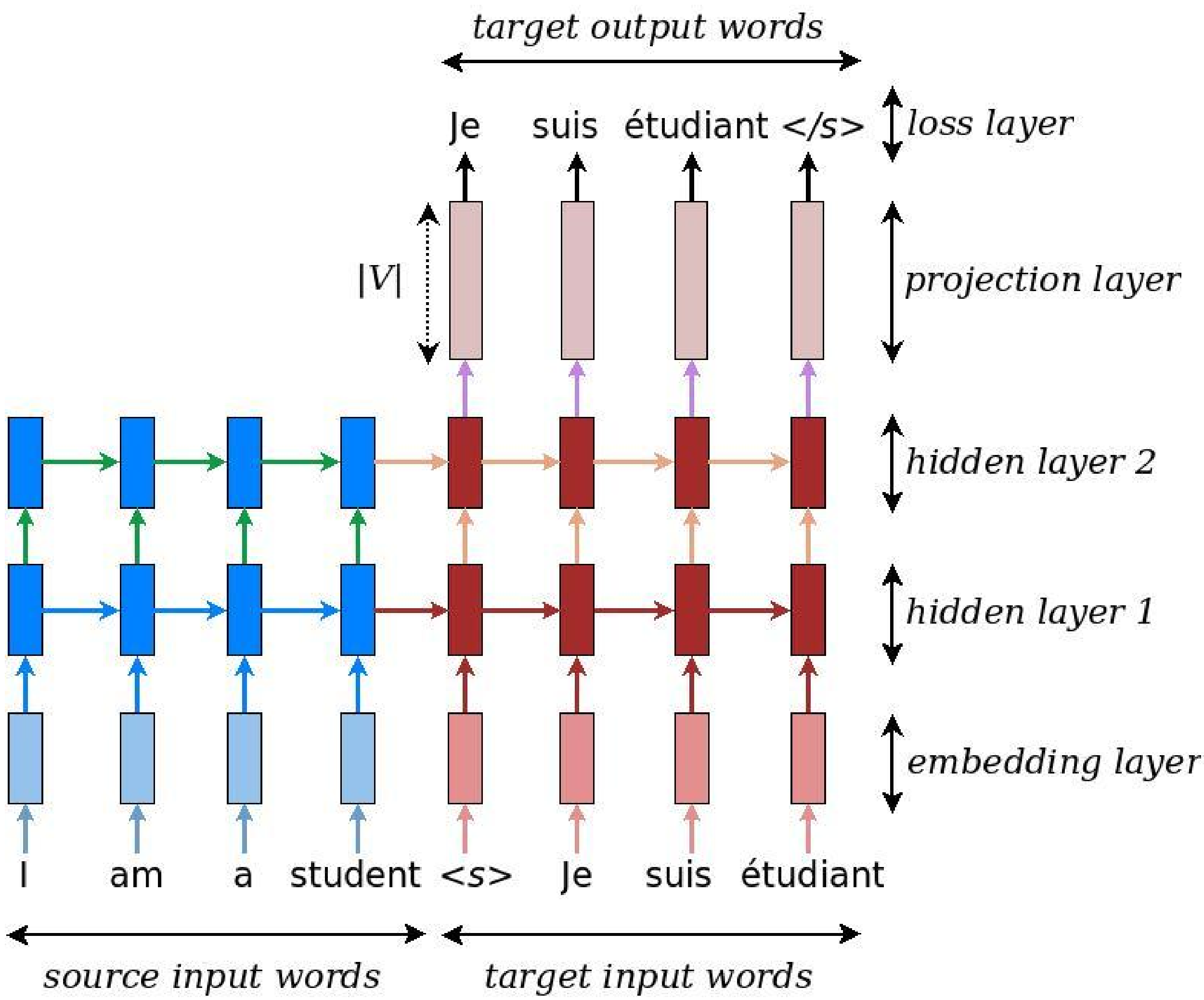


Figure 2. Neural machine translation

数据输入输出shape

以单向、多层LSTM的encoder-decoder模型为例，数据的输入/输出shape为：

- **encoder_inputs** [max_encoder_time, batch_size]: source input words
- **decoder_inputs** [max_decoder_time, batch_size]: target input words

- **decoder_outputs** [max_decoder_time, batch_size]: target output words

- **优点**： 利用了语义信息；能捕获较长依赖信息；翻译流利；
- **缺点**： 无法捕获更长的依赖；训练时间长；
- **类型**：
  - RNN's endocer-decoder
    - unidirectional/bidirectional rnn
    - depth - single/multi layer
    - type - vanilla RNN/LSTM/GRU(LSTM Networks)
  - attention machinism
    - 利用encoder的所有outputs，根据当前 w_t, h_t 计算每个encoder output 对应的权重，然后将encoder outputs加权求和，作为contex vector。

2.2 基于注意力机制的(self attention)

参考李宏毅课程PPT

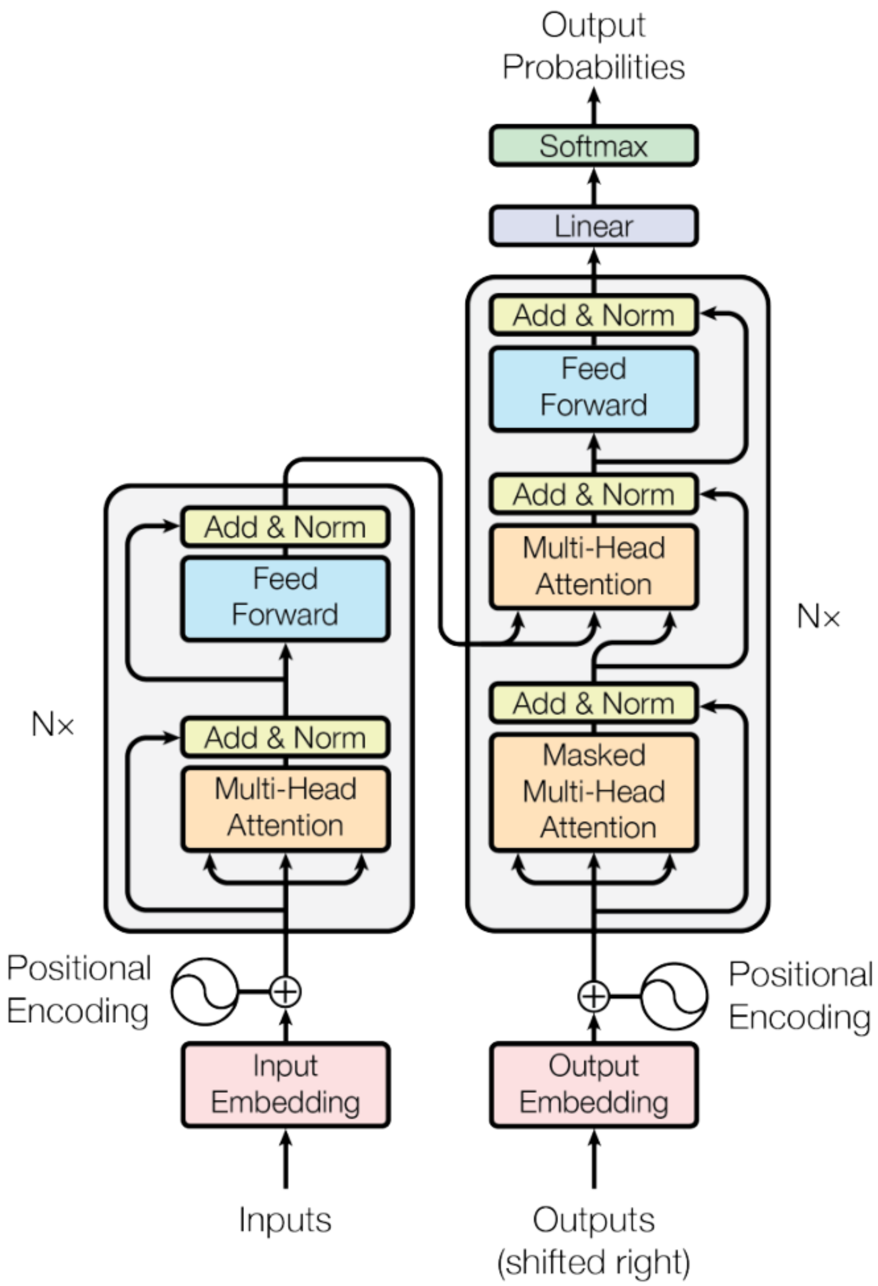


Figure 1: The Transformer - model architecture.

参考文献

词向量相关

1. Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer Science, 2013.
2. Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality[J]. 2013, 26:3111-3119.

词汇编码

1. Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units[J]. arXiv preprint arXiv:1508.07909, 2015.

网络架构

1. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
2. LONG SHORT-TERM MEMORY

LayerNorm

1. Ba J L, Kiros J R, Hinton G E. Layer normalization[J]. arXiv preprint arXiv:1607.06450, 2016.

Seq2Seq

1. Sequence to Sequence Learning with Neural Networks
2. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation
3. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Advances in Neural Information Processing Systems. 2017: 6000-6010.
4. Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

博客/教程

1. https://jalammar.github.io/visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention/
2. https://jalammar.github.io/illustrated-transformer/
3. https://pytorch.org/tutorials/intermediate/seq2seq_translation_tutorial.html#sphx-glr-intermediate-seq2seq-translation-tutorial-py
4. https://github.com/PaddlePaddle/models/tree/develop/PaddleNLP/neural_machine_translation/transformer
5. https://github.com/tensorflow/nmt
6. http://colah.github.io/posts/2015-08-Understanding-LSTMs/
7. https://zhuanlan.zhihu.com/p/54743941
8. https://zhuanlan.zhihu.com/p/49271699