

Определение наилучшего ответа на StackOverflow

Никита Подгузов

Научный руководитель: Рауф Курбанов

Санкт-Петербургский Академический университет

30 марта 2018 года

Возможности сервисов вопросов и ответов:

- Задавать вопрос (и отмечать правильный ответ)
- Отвечать на вопросы, заданные другими пользователями
- Голосовать за понравившиеся ответы

YAHOO! ANSWERS

Quora

 stackoverflow

Особенности системы:

- Узкоспециализированная
- Большая база вопросов
- Наличие сниппетов кода в вопросах и ответах

[Questions](#)
[Developer Jobs](#)
[Tags](#)
[Users](#)

[Log In](#)
[Sign Up](#)

What is the "$x-->$" operator in C++?

▲

7470

After reading [Hidden Features](#) and [Dark Corners of C++/STL](#) on [csapp.Lang.C++.accelerated](#), I was completely surprised that the following snippet compiled and worked in both Visual Studio 2008 and GCC 4.4

Here's the code:

```

#include <stdio.h>
int main()
{
    int x = 10;
    while (x --> 0) // x goes to 0
    {
        printf("Id ", x);
    }
}

```

I'd assume this is C, since it works in GCC as well. Where is this defined in the standard, and where has it come from?

[c++](#)
[operators](#)
[code-formatting](#)
[standards-compliance](#)

share Improve this question

edited Feb 13 '17 at 8:32

community wiki
 28 revs, 20 users 35%
 GManNickG

asked

8 years, 5 months ago

viewed

596,100 times

active

20 days ago

BLOG

☐
[Quantum Computing Site Launches with the Help of Strangeworks](#)

21 Answers

active

oldest

votes

▲

7204

$x-->$ is not an operator. It is in fact two separate operators, $--$ and $>$.

The conditional's code decrements x, while returning x's original (not decremented) value, and then compares the original value with 0 using the $>$ operator.

To better understand, the statement could be written as follows:

```

while( (x--) > 0 )

```

[share](#)
[Improve this answer](#)

edited Aug 9 '15 at 10:22

community wiki
 11 revs, 10 users 15%
 Potatoeswater

▲

2183

That's a very complicated operator, so even ISO/IEC JTC1 (Joint Technical Committee 1) placed its description in two different parts of the C++ Standard.

Joking aside, the two are different operators: $--$ and $>$ described respectively in §5.2.6/2 and §5.9 of the C++03 Standard.

[share](#)
[Improve this answer](#)

edited Mar 2 '17 at 6:49

community wiki
 11 revs, 8 users 33%
 Kon'V. Lyubivskiy

Введение

Постановка задачи

Проблемы:

- Большая доля "неразрешенных" вопросов
- Нет возможности помочь оценить правильность ответов пользователю, задавшему новый вопрос

Хотим научиться определять правильные ответы, используя базу вопросов StackOverflow

Введение

Google & StackOverflow

Google ruby capitalize

About 369.000 results (0,34 seconds)

Capitalize first letter in ruby - Stack Overflow
<https://stackoverflow.com/questions/3724913/capitalize-first-letter-in-ruby>
Sep 16, 2010 - Unfortunately, it is impossible for a machine to upcase/downcase/capitalize properly. ... That's why Ruby's String class only supports capitalization for ASCII characters, because there it's at least somewhat well-defined.

How to convert a string to lower or upper case in Ruby	21 Dec 2014
ruby - Using 'capitalize' or 'capitalize!'	29 Sep 2014
Capitalize the first letter of each word - Ruby	26 Jan 2014
Capitalize only first character of string and leave others alone ...	12 Nov 2011

More results from stackoverflow.com

Class: String (Ruby 2.2.0) - Ruby-Doc.org
<https://ruby-doc.org/core-2.2.0/String.html>
Jump to **capitalize** - capitalize => new_str click to toggle source. Returns a copy of str with the first character converted to uppercase and the remainder to lowercase. Note: case conversion is effective only in ASCII region. "hello".capitalize #=> "Hello" "HELLO".capitalize #=> "Hello" "123ABC".capitalize #=> "123abc".
::try_convert - capitalize! - chomp - gsub

Old version

Google ruby capitalize

Alle Nieuws Shopping Afbeeldingen Video's Meer Instellingen Tools

Ongeveer 370.000 resultaten (0,34 seconden)

Capitalize first letter in ruby - Stack Overflow
<https://stackoverflow.com/questions/3724913/capitalize-first-letter-in-ruby>
The upcase method capitalizes the entire string. I need to capitalize only the first letter. Also, I need to support several popular languages, like German and Russian. How do I do it?

7 antwoorden

Best Answer 197 votes It depends on Ruby version you use. Ruby 2.4 and higher it just works, as since this version ruby supports Unicode case mapping. "wapw".capitalize	Antwoord 2 van 7 37 votes capitalize first letter of first word of string "kirk douglas".capitalize #=> "Kirk douglas" capitalize first letter of each word in rails: "kirk	Antwoord 3 van 7 18 votes Unfortunately, it is impossible for a machine to upcase/downcase/capitalize properly. It needs way contextual information
---	--	--

Class: String (Ruby 2.2.0) - Ruby-Doc.org
<https://ruby-doc.org/core-2.2.0/String.html> - Vertaal deze pagina
Spring naar **capitalize** - capitalize => new_str click to toggle source. Returns a copy of str with the first character converted to uppercase and the remainder to lowercase. Note: case conversion is effective only in ASCII region. "hello".capitalize #=> "Hello" "HELLO".capitalize #=> "Hello" "123ABC".capitalize ...
::try_convert - #[] - count - dump

New version

Введение

Обзор имеющихся решений

"Towards Predicting the Best Answers in CB QAS" (Tian et al. 2013)

- Три вида фичей: $A \leftrightarrow A$, $A \leftrightarrow Q$, A
- Использование Vector Space Model + TF-IDF для определения похожести
- Использование лингвистических фичей (длина текста, количество предложений, читаемость и др.)
- Учитывается лишь наличие/отсутствие сниппетов кода
- Random Forest Classifier

Введение

Обзор имеющихся решений

"State of the art Best Answer Prediction based on Discretisation of Shallow Linguistic Features" (Gkotsis et al. 2014),

"Moving to Stack Overflow: Best-Answer Prediction in Legacy Developer Forums" (Calefato et al. 2016)

- Четыре вида фичей: $A \leftrightarrow A$, A , *user-rating* и *answer-rating*, *thread*
- Использование лингвистических фичей (длина текста, количество предложений, читаемость и др.)
- Использование вероятностной униграммной модели для оценки вероятности ответа
- Использование группировки ответов и дискретизации фичей
- Не учитывает сниппеты кода
- Alternating Decision Tree Classifier

Введение

Минусы имеющихся решений

- Не используется текст вопроса
- Не учитывается порядок слов в предложении
- Не учитываются синонимы и похожие слова, то есть игнорируется семантика
- Не используется содержание сниппетов кода

Цели и задачи

Цель: научиться определять правильность ответа на StackOverflow, используя как его текст, так и код, который может присутствовать внутри ответа

Задачи:

- Реализовать классификатор на основе нейронных сетей, использующий текст ответов
- Добавить использование сниппетов кода в классификаторе
- Сравнить результаты с имеющимися работами
- Проанализировать влияние наличия фичей от сниппетов кода на точность классификации

Данные

Общие факты

Данные:

- Дамп базы вопросов StackOverflow
- XML-файл размером $\sim 50GB$
- Новые данные могут быть получены с помощью API
- Формат файла: *type_id, id, score, date, body*

Данные

Анализ и обработка

Анализ:

- 40 миллионов постов, из них 16 миллионов вопросов и 24 миллионов ответов
- 7 миллионов вопросов (47%) без отмеченного правильного ответа
- 2 миллиона вопросов (13%), у которых нет ни одного ответа
- 21.5 миллионов постов (54%), в которых присутствуют сниппеты кода.

Обработка:

- Удаляем вопросы с рейтингом ≤ 0 , а также вопросы, у которых нет ни одного ответа
- Из остальных постов извлекаем его *body*
- Сохраняем весь код, находящийся в тегах `<code>`
- Очищаем от тегов и сохраняем весь остальной текст вопроса/ответа

После обработки получили 6.5 миллионов вопросов, из них 2 миллиона вопросов (31%) без правильного ответа, а также 13.5 миллионов ответов

Подходы к анализу текста

Bag of words

Идея:

- Каждому слову сопоставляем вектор длины, равной размеру словаря
- Документ: сумма векторов слов

Проблемы:

- Не учитывается семантика
- Не учитывается порядок слов
- Большая размерность

Подходы к анализу текста

Bag of words (пример)

Document 1

The quick brown fox jumped over the lazy dog's back.

Document 2

Now is the time for all good men to come to the aid of their party.

Term	Document 1	Document 2
aid	0	1
all	0	1
back	1	0
brown	1	0
come	0	1
dog	1	0
fox	1	0
good	0	1
jump	1	0
lazy	1	0
men	0	1
now	0	1
over	1	0
party	0	1
quick	1	0
their	0	1
time	0	1

Stopword List

for
is
of
the
to

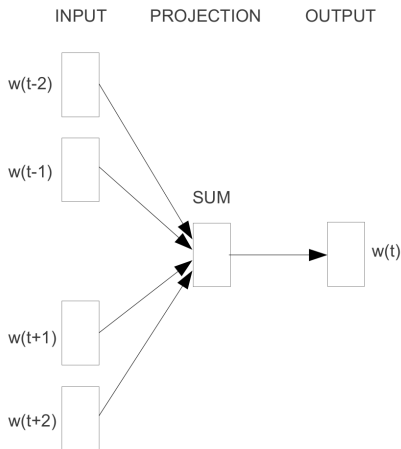
Подходы к анализу текста

Word2Vec

Идея:

- Каждому слову сопоставляем вектор фиксированной длины
- Обучаем на неразмеченном корпусе текстов CBOW/Skip-gram архитектуру
- Вектор отражает смысл слова, сохраняется семантика

Чтобы учесть специфику технического языка, обучаться лучше на текстах `StackOverflow`



CBOW

Подходы к анализу текста

Рекуррентные нейронные сети

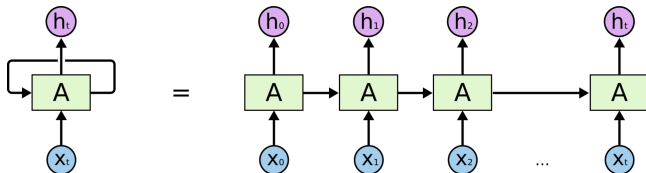
Хотим научиться учитывать порядок слов в тексте

Идея:

- Используем embedding слов из Word2Vec
- Отдаем на вход клетке сети новое слово и выход с предыдущей (учет контекста)
- Хотим, чтобы выход сети отражал смысл входного текста

Одна из двух основных архитектур: *LSTM*-клетка

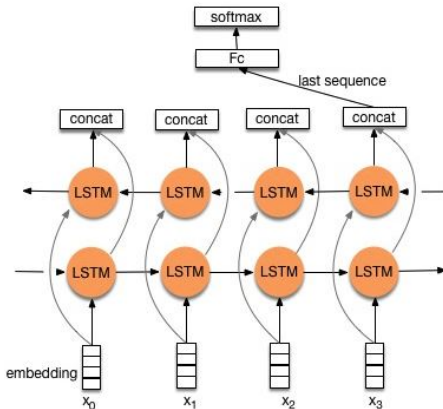
Также используются двунаправленные рекуррентные нейронные сети для захвата контекста справа



Классификация текстов

Базовая архитектура

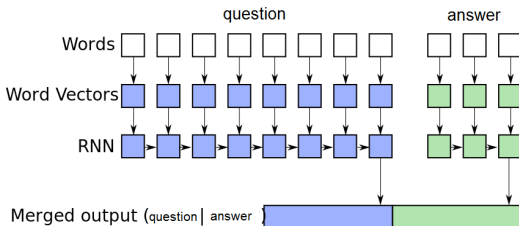
- Классификация ответов на два класса: правильный/неправильный
- В качестве embedding-а используется векторное представление, обученное на текстах вопросов и ответов StackOverflow



Классификация ответов

Учет текста вопроса

- Хотим также учитывать текст вопроса, чтобы понимать релевантность ответа
- Добавим *BiLSTM*-сеть для текста вопроса и будем использовать эти признаки вместе с признаками ответа



Подходы к анализу кода

Проблемы

Код похож на текст, поэтому можно попробовать применить аналогичные методы

Проблема: нелинейная структура кода (циклы, ветвления и др.)

Замечание: тем не менее, может хорошо сработать для однострочных сниппетов

Подходы к анализу кода

Использование синтаксического дерева

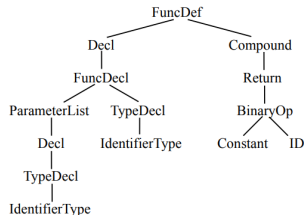
(Mou et al., 2014)

Идея:

- Вместо текста кода рассмотрим его синтаксическое дерево
- Обучаем Code2Vec, используя в качестве контекста сыновей в синтаксическом дереве
- Нормализуем вектора сыновей по размеру их поддерева

```
double doubles(double doublee){  
    return 2 * doublee;  
}
```

A C code snippet



The corresponding AST

Подходы к анализу кода





Использование метода

- Обучаем модель на корпусе кода фиксированного языка программирования (например, Python)
- В качестве embedding-ов вершин синтаксического дерева используем полученные векторные представления
- Используем RNN, как в случае текста, отдавая на вход полученные embedding-и в порядке обхода *dfs*-ом

Выводы

Результаты

- *сравнение с имеющимися решениями*
- Анализ того, какие текстовые представления работают лучше всего
- Анализ того, как улучшилась классификация после добавления учета содержания сниппетов кода

- 1  [Tian et al. \(2013\)](#)
Towards Predicting the Best Answers in Community-Based Question-Answering Services
- 2  [Gkotsis et al. \(2014\)](#)
It's all in the Content: State of the art Best Answer Prediction based on Discretisation of Shallow Linguistic Features
- 3  [Calefato et al. \(2016\)](#)
Moving to Stack Overflow: Best-Answer Prediction in Legacy Developer Forums
- 4  [Mou et al. \(2014\)](#)
Building Program Vector Representations for Deep Learning