

Can we NLP a title?

Elijah Berenstein-Cooper, Ahmed Saif, Ben Conrad

November 6, 2014

1 Introduction

Under the context of natural language processing, this lab explores the relation between job descriptions and salaries. This topic was the focus of a Kaggle competition whose sponsor, Adzuna¹, had a database of job listings of which only half provided salary information (the winner recieved \$3000). As applicants will more likely apply to descriptions that give a salary, Adzuna's placement rate (and hence revenue) is improved if they can provide an estimated salary for those descriptions that did not originally include one. (The employee recruiting business is structured so that Adzuna generally can't directly ask the companies to provide salary estimates.) This is challenging from the legal standpoint, as grossly incorrect salaries may expose Adzuna to claims from applicants and companies, and applicant experience, since Adzuna's estimates must seem plausible to applicants before they will be willing to spend the time applying.

While Adzuna could manually estimate these salaries, scalability encourages throwing computers at the problem. In this lab we will be using Adzuna's job description and salary datasets, divided into training and test sets. These descriptions vary in word count, industry, employment level, and company location, while the salaries are the mean of the provided salary range. The variability in description content leads to a notoriously sparse matrices, so we will be interested in the tradeoffs of various feature descriptors. The naive approach to this problem is to count the occurrences of individual words and associate them to salaries; here each word is a feature and as there are many descriptive words the resulting matrices will be sparse. Other feature choices may be individual word length, occurrences of word pairs or triplets (ie 'technical communication'), n-grams (sequences of n characters), and many others. Note that it is common to ignore stop words like 'the', 'a', 'it', 'you', 'we', etc... because they add little information.

2 Warm-Up

Here are two examples from the dataset:

Engineering Systems Analyst Dorking Surrey Salary ****K Our client is located in Dorking, Surrey and are looking for Engineering Systems Analyst our client provides specialist software development Keywords Mathematical Modelling, Risk Analysis, System Modelling, Optimisation, MISER, PIONEER Engineering Systems Analyst Dorking Surrey Salary ****K

with a salary of \$25,000 and

A subsea engineering company is looking for an experienced Subsea Cable Engineer who will be responsible for providing all issues related to cables. They will need someone who has at least 1015 years of subsea cable engineering experience with significant experience within offshore oil and gas industries. The qualified candidate will be responsible for developing new modelling methods for FEA and CFD. You will also be providing technical leadership to all staff therefore you must be an expert in problem solving and risk assessments. You must also be proactive and must have strong interpersonal skills. You must be a Chartered Engineer or working towards it the qualification. The company offers an extremely competitive salary, health care plan, training, professional membership sponsorship, and relocation package

having a salary of \$85,000.

We'll first apply the word count feature descriptor (first 11 shown alphabetically):

Description 1		Description 2	
****k	2	1015	1
analysis	1	a	2
analyst	3	all	2
and	1	also	2
are	1	an	3
client	2	and	5
development	1	assessments	1
dorking	3	at	1
engineering	3	be	6
for	1	cable	2
in	1	cables	1

We can collect these word counts into the matrix A, and the salaries into the vector b:

$$A = \begin{bmatrix} 2 & 1 & 3 & 1 & 1 & 2 & 1 & 3 & 3 & 1 & 1 \cdots \\ 1 & 2 & 2 & 2 & 3 & 5 & 1 & 1 & 6 & 2 & 1 \cdots \end{bmatrix}$$

$$b = \begin{bmatrix} 2500 \\ 85000 \end{bmatrix}$$

The least-squares solution to this problem, x is:

$$\hat{x} = \begin{bmatrix} 1819.6517 \\ 1730.9558 \\ 1427.6805 \\ 1250.2887 \\ 1213.1011 \\ 1213.1011 \\ 1213.1011 \\ 909.8258 \\ 909.8258 \\ 909.8258 \\ 732.4341 \\ \vdots \end{bmatrix} \begin{array}{l} \text{"be" from}[0, 6] \\ \text{"and" from}[1, 5] \\ \text{"for" from}[1, 4] \\ \text{"engineering" from}[3, 2] \\ \text{"must" from}[0, 4] \\ \text{"will" from}[0, 4] \\ \text{"you" from}[0, 4] \\ \text{"an" from}[0, 3] \\ \text{"subsea" from}[0, 3] \\ \text{"the" from}[0, 3] \\ \text{"modelling" from}[2, 1] \\ \vdots \end{array}$$

For two samples, it should not be surprising that the most heavily-weighted words are unique to each description.