

# Armenian Job Market

*David Gadyan*

*September 10, 2018*

As a matter of fact with the rise of technological era, old methods of job advertisement such as newspapers, leaflets and other paper based means of communication step down to more advanced and sophisticated technological tools such as online job posting websites, employment agencies using advanced technological tools etc. In this abundance of choices and ever simplifying methods of posting jobs online, more and more companies turn to online job posting websites. This leads to the situation when data on the Web becomes more and more representative of real economical demand for jobs. This in turn reveals new prospects to analyze job market patterns applying state-of-the-art programmes and methods to get data from Web and analyze it. This presents great opportunities for automatization and optimization as we can data online by no means and conduct analysis in very short period of time even automatizing code to extend that we may one day open a website or a dashboard and see all useful info without any hard coding and etc. Nevertheless, in this analysis we did not have goal to automate it to that extend but in near future in my view this will be also done.

Our analysis will be centered around Armenian Job Market and particularly IT sector for the period from 2005 to 2018 years. At the time of conducting this analysis 2018 has not finished yet and 2018's data represents months from January to August.

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.2.1

## v ggplot2 3.0.0      v purrr   0.2.5
## v tibble  1.4.2      v dplyr  0.7.6
## v tidyr   0.8.1      v stringr 1.3.1
## v readr   1.1.1      v forcats 0.3.0

## -- Conflicts ----- tidyverse_conflicts()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
library(tm)
```

```
## Loading required package: NLP
```

```
##
```

```
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
```

```
##
```

```
##      annotate
```

```
library(dplyr)
```

```
library(formattable)
```

```
library(formatR)
```

```
library(knitr)
```

```
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
```

The data was manually scrapped from the following website.[link](#).

```
jobs_2005_2018 <- read.csv("C:\\Users\\Gaya\\Desktop\\R\\Workings\\Armenian Job Market\\df_final5.csv",
  stringsAsFactors = F)
jobs_2005_2018[, 1] <- NULL
jobs_2005_2018$opening_date <- as.Date(jobs_2005_2018$opening_date,
  "%Y-%m-%d")
jobs_2005_2018$application_deadline_date <- as.Date(jobs_2005_2018$application_deadline_date,
  "%Y-%m-%d")
```

Let's look at the structure of our dataframe which was scraped manually from one an Armenian online job posting website. As we see there are 24,288 observations with 11 variables.

```
str(jobs_2005_2018)
```

```
## 'data.frame': 24288 obs. of 11 variables:
## $ company : chr "M-possible" "Lycos Europe" "Lycos Europe" "Lycos Europe" ...
## $ title : chr "Senior Software Engineer" "Technical System Documenter" "Flash A
## $ term : chr NA NA NA NA ...
## $ duration : chr NA "Permanent" "Permanent" "Permanent" ...
## $ location : chr "Yerevan, Armenia" "Yerevan, Armenia" "Yerevan, Armenia" "Yerevan
## $ job_responsibilities : chr "Actively identify and implement tools, resources, new technologi
## $ required_qualifications : chr "Minimum 3 years of experience in game industry; MS in Computer S
## $ opening_date : Date, format: "2004-12-23" "2004-12-23" ...
## $ application_deadline_date: Date, format: "2005-01-31" "2005-01-15" ...
## $ job_id : int 1699 1700 1701 1702 1703 1704 1706 1707 1708 1709 ...
## $ industry : chr "Information Technology" "Information Technology" "Information Te
```

```
head(jobs_2005_2018)
```

```
##      company      title term duration
## 1 M-possible Senior Software Engineer <NA> <NA>
## 2 Lycos Europe Technical System Documenter <NA> Permanent
## 3 Lycos Europe Flash Action Scripting Programmer <NA> Permanent
## 4 Lycos Europe Cartoon Artist <NA> Permanent
## 5 Lycos Europe Mathematician System Analyzer <NA> Permanent
## 6 Lycos Europe Photoshop Graphics Web Designer <NA> Permanent
##      location
## 1 Yerevan, Armenia
## 2 Yerevan, Armenia
## 3 Yerevan, Armenia
## 4 Yerevan, Armenia
## 5 Yerevan, Armenia
## 6 Yerevan, Armenia
##
## 1
## 2
## 3
## 4
## 5 Your job will be to make the formulas and excel\nsheets necessary to watch the economics and flow
## 6
##
## 1 Minimum 3
## 2
## 3 Expert in using Macromedia Flash MX 2004 Version 7; Expert in programming Flash Action Scripting 1
## 4
## 5
```

```
## 6                                     Expert in Adobe Photoshop; Expert in User i
## opening_date application_deadline_date job_id industry
## 1 2004-12-23 2005-01-31 1699 Information Technology
## 2 2004-12-23 2005-01-15 1700 Information Technology
## 3 2004-12-23 2005-01-15 1701 Information Technology
## 4 2004-12-23 2005-01-15 1702 Information Technology
## 5 2004-12-23 2005-01-15 1703 Education & Teaching
## 6 2004-12-23 2005-01-15 1704 Information Technology
```

As we may observe we have two date type variables `opening_date` and `application_deadline_date` let's analyze them to get Average Hiring Duration to compare it with other countries' results. To do this we need to separate 2017 year data.

```
jobs_2017 <- jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
"%Y")) %>% filter(Year %in% c(2017))
```

To get the Average Hiring Duration in Armenia during 2017 we extract `opening_date` from `application_deadline_date` and take the mean.

```
av_time_open_deadline_final <- mean(jobs_2017$application_deadline_date -
jobs_2017$opening_date)
av_time_open_deadline_final
```

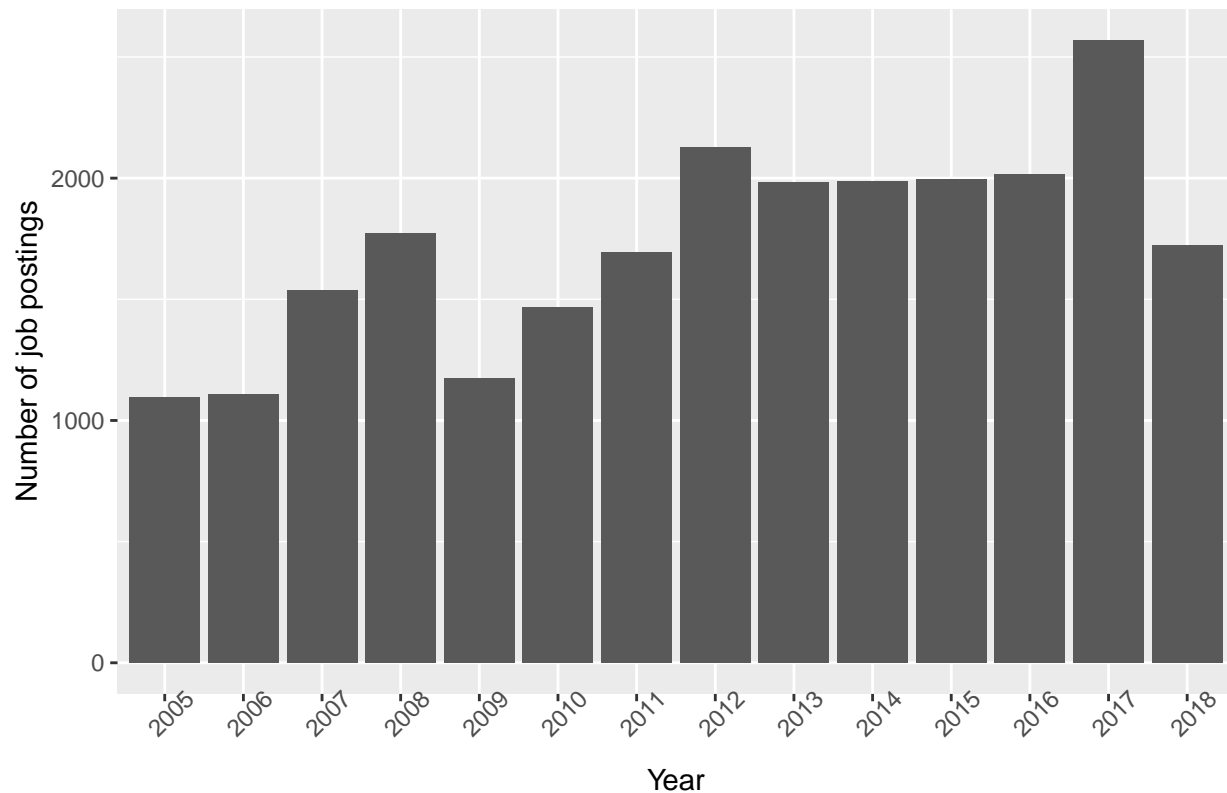
## Time difference of 23.62787 days

The Average Hiring Duration in Armenia during 2017 was 23.6 days. Now let's compare it with worldwide result published by **glassdoor** which is highly respected company in the Job Market. According to **glassdoor** the average length of job interview processes in 2017 was 23.7 days across all 25 countries in their sample. You can find this study through the following link **link**. As we may observe Armenia's Average Hiring Duration is practically the same as global average, which is a good result.

Now let's explore the proportion of jobs in each industry.

```
jobs_2005_2018 %>% group_by(format(jobs_2005_2018$opening_date,
"%Y")) %>% summarise(count = n()) %>% filter(`format(jobs_2005_2018$opening_date, "%Y")` >
2004) %>% ggplot(aes(x = `format(jobs_2005_2018$opening_date, "%Y")`,
y = count)) + geom_bar(stat = "identity") + labs(title = "Chart1. Historical Job Posting in Armenia :
x = "Year", y = "Number of job postings") + theme(axis.text.x = element_text(angle = 45))
```

Chart1. Historical Job Posting in Armenia 2005–2018\*



As we may easily observe the online job openings had doubled from 2005 to 2017. The number of postings peaked in 2017. Nevertheless, in 2018 the number of job openings was lower compared to 2017. We will look closer to this situation because we cannot conclude conclusively looking only on the above barplot Chart1. The primary reason for the previous statement is that 2018 year does not include full year data as year did not ended at the time of this analysis.

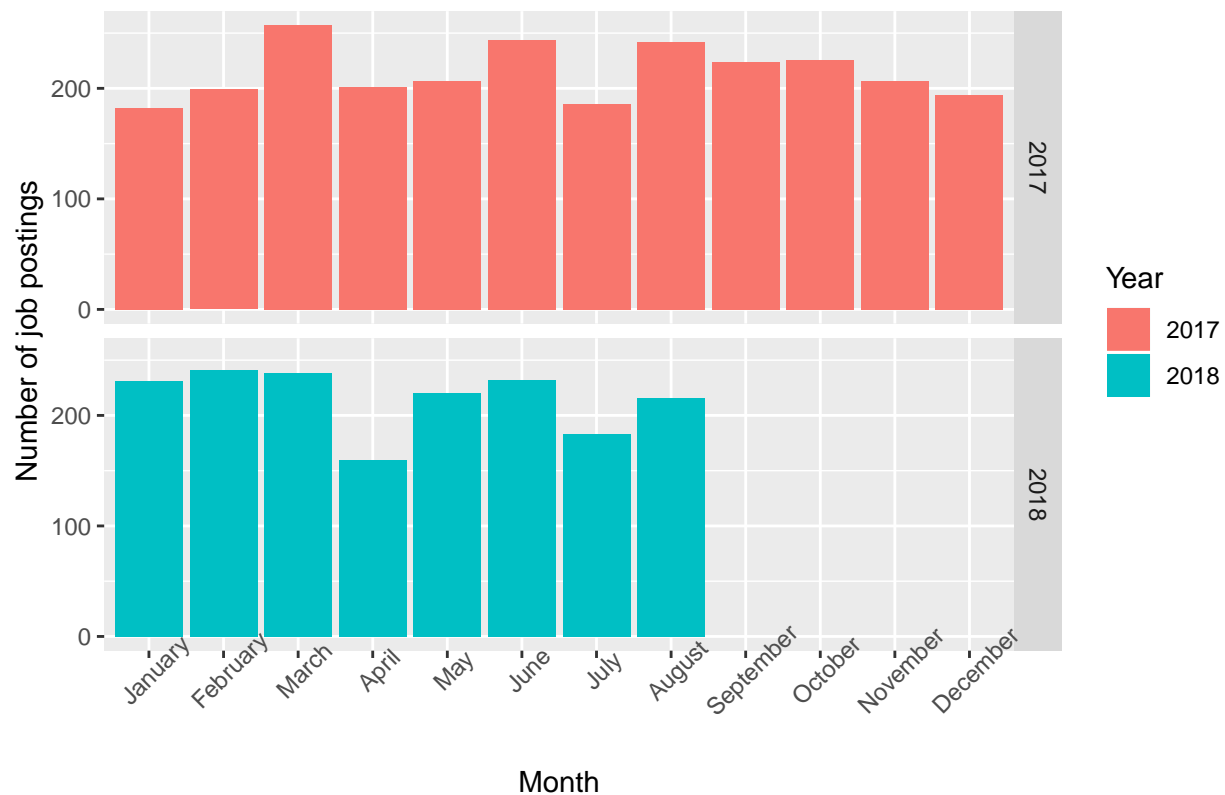
```
max(jobs_2005_2018$opening_date)
```

```
## [1] "2018-08-31"
```

As we can observe the latest date is 2018-08-31. So we will visualize in our next graph 2017 and 2018 from years start and until this date.

```
jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
"%Y"), Month = format(jobs_2005_2018$opening_date, "%B")) %>%
filter(Year %in% c(2017, 2018)) %>% group_by(Year, Month) %>%
summarise(count = n()) %>% arrange(match(Month, month.name)) %>%
ggplot(aes(x = factor(Month, levels = month.name), y = count,
fill = Year)) + geom_bar(stat = "identity") + labs(title = "Chart2. Job Posting in Armenia 2017-2018",
x = "Month", y = "Number of job postings") + theme(axis.text.x = element_text(angle = 45)) +
facet_grid(Year ~ .)
```

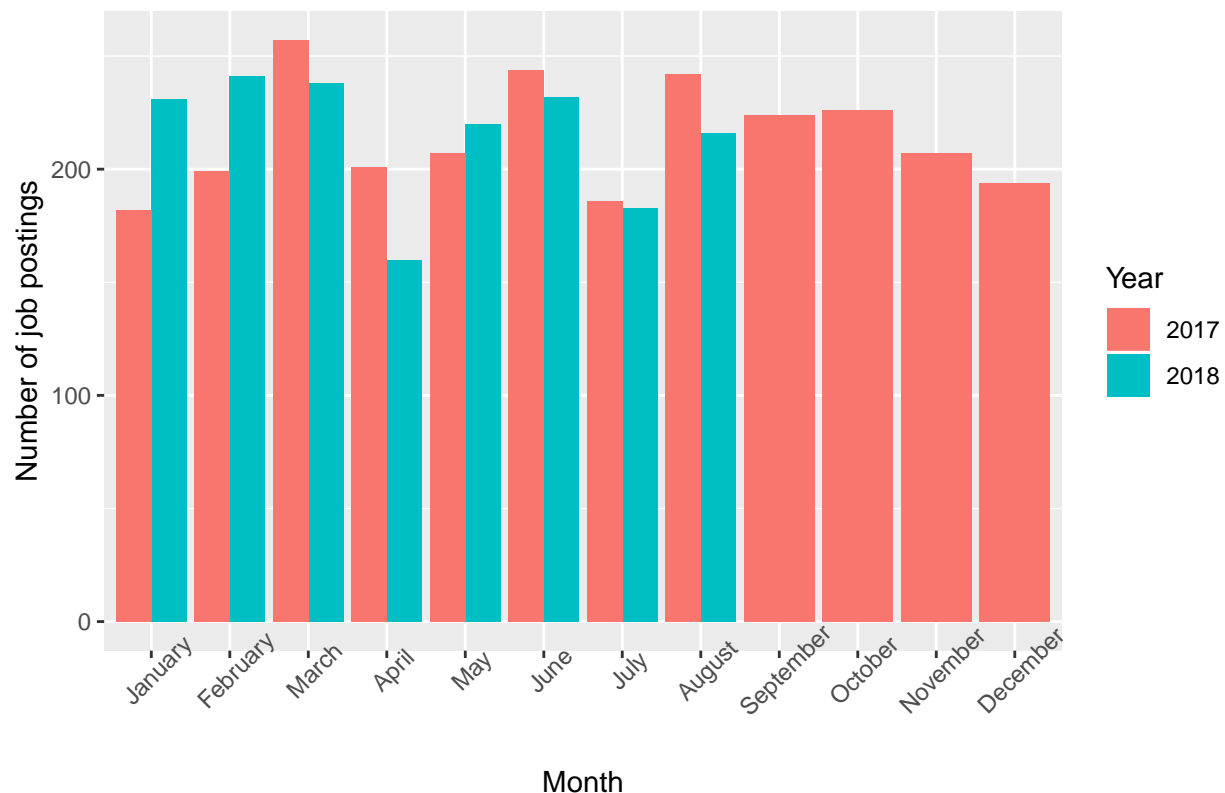
Chart2. Job Posting in Armenia 2017–2018\*



As we stated previously 2018 year does not represent full year. It only includes period from January to August. Despite this we may observe some decline in the number of job postings in 2018 compared to 2017, particularly in April and August. Let's further discover this two years.

```
jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
"%Y"), Month = format(jobs_2005_2018$opening_date, "%B")) %>%
filter(Year %in% c(2017, 2018)) %>% group_by(Year, Month) %>%
summarise(count = n()) %>% arrange(match(Month, month.name)) %>%
ggplot(aes(x = factor(Month, levels = month.name), y = count,
fill = Year)) + geom_bar(stat = "identity", position = "dodge") +
labs(title = "Chart3. Job Posting in Armenia 2017-2018*",
x = "Month", y = "Number of job postings") + theme(axis.text.x = element_text(angle = 45))
```

Chart3. Job Posting in Armenia 2017–2018\*



The above graph puts side by side 2017 and 2018 years and makes comparison between them easier. Chart3 supported our preliminary observation during the June, July and August number of online job postings reduced despite the fact that in 2018's first two months clearly dominated over the same period.

Now let's explore by industry and year.

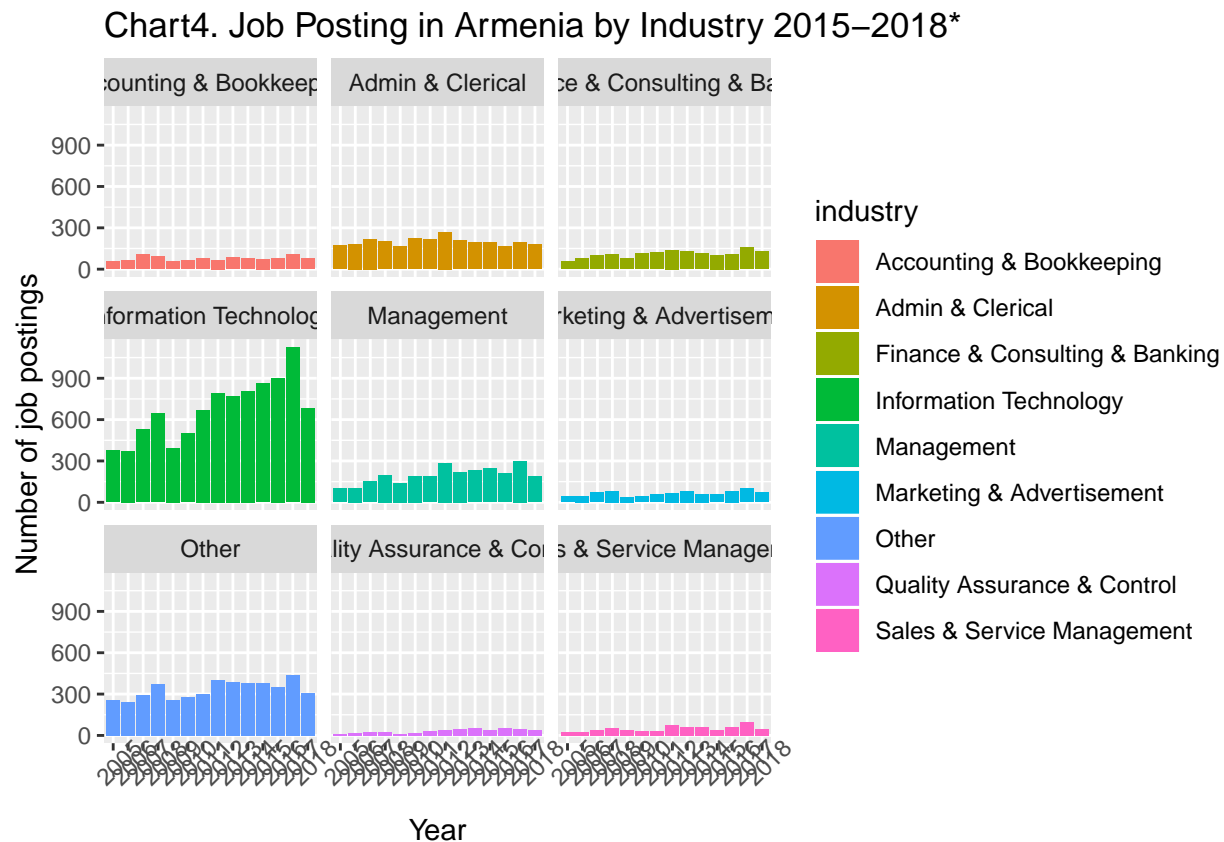
```
colnames(jobs_2005_2018)
```

```
## [1] "company"          "title"
## [3] "term"             "duration"
## [5] "location"         "job_responsibilities"
## [7] "required_qualifications" "opening_date"
## [9] "application_deadline_date" "job_id"
## [11] "industry"
```

```
jobs_2005_2018_s <- data.frame(jobs_2005_2018)
jobs_2005_2018_s[jobs_2005_2018_s$industry %in% c("Construction",
"Customer Service", "Design", "Education & Teaching", "General Labor",
"Health Care", "Human Resources", "Insurance", "Journalism",
"Legal", "Nonprofit & Social Services", "Other", "Real Estate",
"Telecommunications", "Tourism"), ]$industry <- "Other"
```

```
jobs_2005_2018_s %>% mutate(Year = format(jobs_2005_2018$opening_date,
"%Y"), Month = format(jobs_2005_2018$opening_date, "%B")) %>%
filter(Year %in% c(2005, 2006, 2007, 2008, 2009, 2010, 2011,
2012, 2013, 2014, 2015, 2016, 2017, 2018)) %>% group_by(Year,
Month, industry) %>% summarise(count = n()) %>% arrange(match(Month,
month.name)) %>% ggplot(aes(x = Year, y = count, fill = industry)) +
```

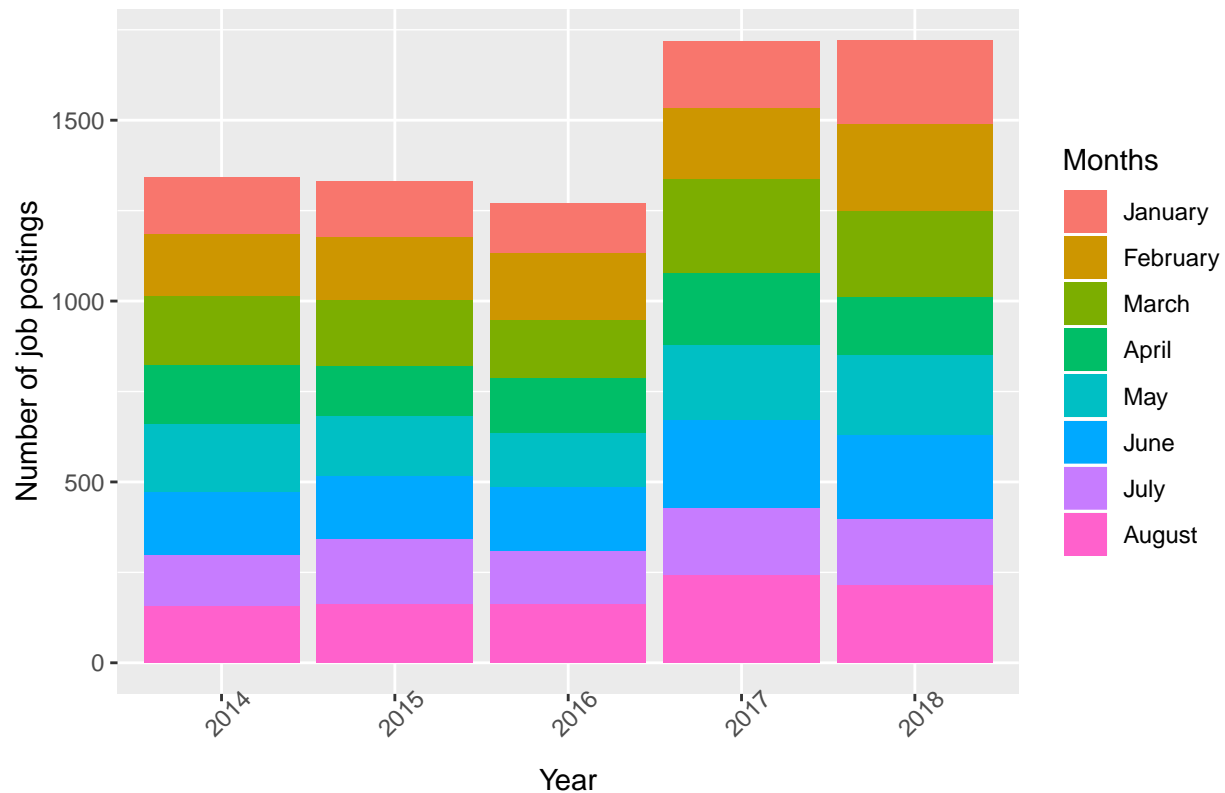
```
geom_bar(stat = "identity") + labs(title = "Chart4. Job Posting in Armenia by Industry 2015-2018*",
x = "Year", y = "Number of job postings") + theme(axis.text.x = element_text(angle = 45)) +
facet_wrap(industry ~ .)
```



As we may easily observe the biggest proportion in our dataset takes Information Technology industry or job category other way around. This is very interesting pattern in our online job postings dataset and we will conduct deeper analysis especially into this job sector further in our analysis.

```
jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
"%Y"), Month = format(jobs_2005_2018$opening_date, "%B")) %>%
filter(Year %in% c(2014, 2015, 2016, 2017, 2018)) %>% group_by(Year,
Month, industry) %>% summarise(count = n()) %>% arrange(match(Month,
month.name)) %>% filter(Month %in% c("January", "February",
"March", "April", "May", "June", "July", "August")) %>% ggplot(aes(x = Year,
y = count, fill = factor(Month, levels = month.name))) +
geom_bar(stat = "identity") + labs(title = "Chart5. Job Posting in Armenia by Industry 2014-2018 fr
x = "Year", y = "Number of job postings") + theme(axis.text.x = element_text(angle = 45)) +
guides(fill = guide_legend(title = "Months"))
```

Chart5. Job Posting in Armenia by Industry 2014–2018 from January–Aug

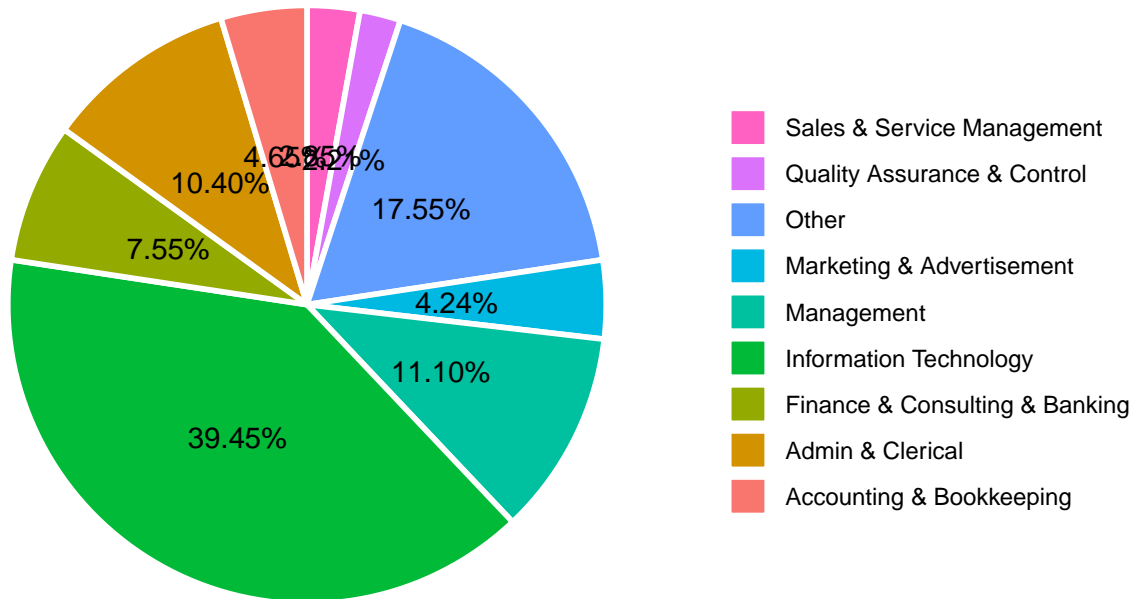


The Chart5 perfectly presents situation in terms of whether 2018 was lower in the number of job postings. As we stated before 2018 year data was not a full as it included months from January-August. It is obvious that in 2018 in first 8 months there was any decline and in the number of online job postings and results at comparable periods are pretty similar even with 2018 in January and February clearly leading in the number of jobs. Nevertheless this results stalled in March and especially in April. The timing follows the Armenian Velvet Revolution which had taken place during April and clearly people and businesses were generally consumed by political activities rather than mere business market.

```
jobs_2005_2018_s %>% mutate(Year = format(jobs_2005_2018_s$opening_date,
"%Y")) %>% filter(Year %in% c(2018)) %>% group_by(Year, industry) %>%
summarise(count = n()) %>% mutate(per = count/sum(count)) %>%
arrange(desc(per)) %>% ggplot(aes(x = "", y = per, fill = industry)) +
geom_bar(width = 1, size = 1, color = "white", stat = "identity") +
coord_polar("y", start = 0) + geom_text(aes(label = percent(per)),
position = position_stack(vjust = 0.5)) + labs(x = NULL,
y = NULL, fill = NULL, title = "Chart6. Jobs by categories 2018") +
guides(fill = guide_legend(reverse = TRUE)) + theme_classic() +
theme(axis.line = element_blank(), axis.text = element_blank(),
axis.ticks = element_blank(), plot.title = element_text(hjust = 0.5,
color = "#666666"))
```



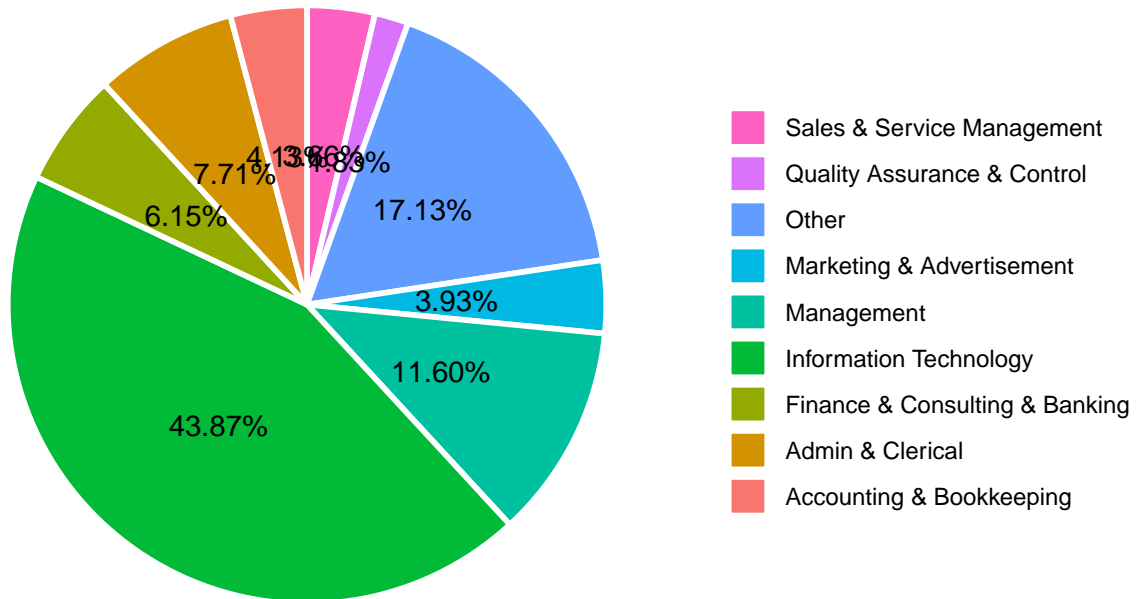
Chart6. Jobs by categories 2018



As we clearly observe the IT industry represents the biggest proportion of all online job postings. The second in the number of job postings is Management category. Let's observe whether the same holds true in 2017 year.

```
jobs_2005_2018_s %>% mutate(Year = format(jobs_2005_2018_s$opening_date,
"%Y")) %>% filter(Year %in% c(2017)) %>% group_by(Year, industry) %>%
summarise(count = n()) %>% mutate(per = count/sum(count)) %>%
arrange(desc(per)) %>% ggplot(aes(x = "", y = per, fill = industry)) +
geom_bar(width = 1, size = 1, color = "white", stat = "identity") +
coord_polar("y", start = 0) + geom_text(aes(label = percent(per)),
position = position_stack(vjust = 0.5)) + labs(x = NULL,
y = NULL, fill = NULL, title = "Chart7. Jobs by categories 2017") +
guides(fill = guide_legend(reverse = TRUE)) + theme_classic() +
theme(axis.line = element_blank(), axis.text = element_blank(),
axis.ticks = element_blank(), plot.title = element_text(hjust = 0.5,
color = "#666666"))
```

Chart7. Jobs by categories 2017



As we see in 2017 the proportion of IT was even higher around 43.87% compared to 2018 39.45% i.e. fell by approximately 4.4%. Now let's define what job title comprises IT industry in order to define the most demanded IT job in Armenia.

```
it_titles_2017 <- jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
"%Y")) %>% filter(Year %in% c(2017) & industry %in% c("Information Technology")) %>%
  group_by(title) %>% summarise(count = n()) %>% mutate(per = count/sum(count)) %>%
  arrange(desc(per))

it_titles_2017 <- data.frame(it_titles_2017)

it_titles_2017[it_titles_2017$count > 3, ]
```

##	title	count	per
## 1	Web Developer	20	0.017746229
## 2	.NET Developer	18	0.015971606
## 3	Digital Marketing Specialist	18	0.015971606
## 4	IT Project Manager	14	0.012422360
## 5	PHP Developer	13	0.011535049
## 6	Senior Front-End Developer	13	0.011535049
## 7	Android Developer	12	0.010647737
## 8	Database Developer	12	0.010647737
## 9	System Administrator	11	0.009760426
## 10	UI/ UX Designer	11	0.009760426
## 11	iOS Developer	10	0.008873114
## 12	Senior Android Developer	10	0.008873114
## 13	Senior Java Developer	10	0.008873114

## 14	Senior PHP Developer	10	0.008873114
## 15	Backend Developer	9	0.007985803
## 16	Front-End Developer	9	0.007985803
## 17	IT Specialist	9	0.007985803
## 18	JavaScript Developer	9	0.007985803
## 19	Java Developer	8	0.007098492
## 20	Back-End Developer	7	0.006211180
## 21	Senior .NET Developer	7	0.006211180
## 22	Network Engineer	6	0.005323869
## 23	Technical Support Specialist	6	0.005323869
## 24	Data Analyst	5	0.004436557
## 25	Full Stack Developer	5	0.004436557
## 26	Full Stack Web Developer	5	0.004436557
## 27	Junior Software Developer	5	0.004436557
## 28	Senior Back-End Developer	5	0.004436557
## 29	Senior C# Developer	5	0.004436557
## 30	Senior JavaScript Developer	5	0.004436557
## 31	Senior Software Developer	5	0.004436557
## 32	Software Developer	5	0.004436557
## 33	Software Engineer	5	0.004436557
## 34	Technical Support Engineer	5	0.004436557
## 35	UI Developer	5	0.004436557
## 36	.NET Service Engineer/ Analyst	4	0.003549246
## 37	Back-end Developer	4	0.003549246
## 38	Business Software Consultant	4	0.003549246
## 39	Customer Attraction Specialist	4	0.003549246
## 40	Full Stack .NET Developer	4	0.003549246
## 41	IT Auditor	4	0.003549246
## 42	QA Automation Engineer	4	0.003549246
## 43	Senior iOS Developer	4	0.003549246
## 44	Senior Software Engineer	4	0.003549246
## 45	Web Content Manager	4	0.003549246

From this list we may see that Web Developer, .NET Developer and Digital Marketing Specialists form respectively the first three positions in the list. However, only looking on this table will not reveal significant insights and as it is known better way is visualization. As saying goes 'it is better one time show than 100 say'.

```
it_titles_2018 <- jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
"%Y")) %>% filter(Year %in% c(2018) & industry %in% c("Information Technology")) %>%
  group_by(title) %>% summarise(count = n()) %>% mutate(per = count/sum(count)) %>%
  arrange(desc(per))

it_titles_2018 <- data.frame(it_titles_2018)

it_titles_2018[it_titles_2018$count > 3, ]
```

##	title	count	per
## 1	Database Developer	10	0.014727541
## 2	.NET Developer	9	0.013254786
## 3	Android Developer	9	0.013254786
## 4	Software Engineer	9	0.013254786
## 5	System Administrator	9	0.013254786
## 6	Full Stack Developer	8	0.011782032

## 7	iOS Developer	8 0.011782032
## 8	PHP Developer	8 0.011782032
## 9	Full Stack .NET Developer	7 0.010309278
## 10	Senior .Net Developer	7 0.010309278
## 11	Senior Java Developer	7 0.010309278
## 12	UI/ UX Designer	7 0.010309278
## 13	Web Developer	7 0.010309278
## 14	.Net Developer	5 0.007363770
## 15	C# Software Developer	5 0.007363770
## 16	Digital Marketing Specialist	5 0.007363770
## 17	IT Specialist	5 0.007363770
## 18	Technical Support Specialist	5 0.007363770
## 19	Automated QA Engineer	4 0.005891016
## 20	Front-end Developer	4 0.005891016
## 21	Java Developer	4 0.005891016
## 22	Network Engineer	4 0.005891016
## 23	Senior .NET Developer	4 0.005891016
## 24	Senior C++ Software Engineer	4 0.005891016
## 25	Senior Software Engineer	4 0.005891016
## 26	Software Developer	4 0.005891016
## 27	Technical Writer	4 0.005891016

```
set.seed(42)
wordcloud(words = it_titles_2018$title, freq = it_titles_2018$count,
  min.freq = 3, scale = c(1.8, 0.1), random.order = F, colors = brewer.pal(8,
    "Dark2"))
```



The wordcloud above presents IT jobs demanded the most in 2018 year by the employers. We see that Database Developers, System Administrators, Android Developers, Full Stack Developers are highly by demanded by Armenian Job Market in 2018 year. We may gain other useful results from the wordcloud above but we leave it to the subjective analysis and needs of the user of this project.

In the following steps we define a function to make our analysis of Most Demanded IT Jobs on a yearly basis.

```
it_wordcloud_f <- function(year = 2018, min_freq = 5) {
  it_title <- jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
    "%Y")) %>% filter(Year %in% c(year) & industry %in% c("Information Technology")) %>%
    group_by(title) %>% summarise(count = n()) %>% mutate(per = count/sum(count)) %>%
    arrange(desc(per))
  layout(matrix(c(1, 2), nrow = 2), heights = c(30, 100))
  par(mar = rep(1, 4))
  plot.new()
  text(x = 0.5, y = 0.5, labels = paste("Most popular IT jobs in ",
    year))
  w <- wordcloud(words = it_title$title, freq = it_title$count,
    min.freq = min_freq, scale = c(1.1, 0.1), random.order = F,
    colors = brewer.pal(8, "Dark2"), main = "Title")
}
```

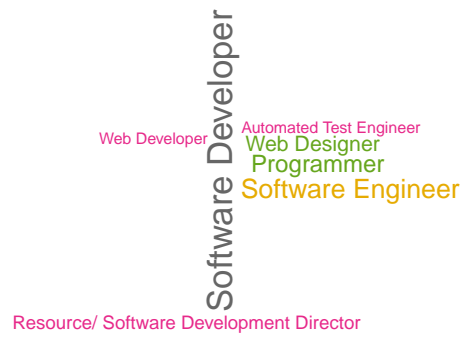
```
it_wordcloud_f(2018)
```

### Most popular IT jobs in 2018



```
for (i in 2005:2018) {
  it_wordcloud_f(i)
}
```

## Most popular IT jobs in 2005



## Most popular IT jobs in 2006



## Most popular IT jobs in 2007





Most popular IT jobs in 2008



## Most popular IT jobs in 2009



Research & Development Engineer  
System Administrator  
Senior Java Developer  
Software Developer  
Technical Consultant

## Most popular IT jobs in 2010



A word cloud showing the most popular IT jobs in 2010. The words are arranged in a cluster, with 'Senior Software Engineer' being the largest and most prominent. Other jobs listed include 'Web Designer', 'Senior Java Developer', 'Technical Writer', and 'Web Developer'.

Job Title	Relative Popularity (Size)
Senior Software Engineer	High
Web Designer	Medium
Senior Java Developer	Medium
Technical Writer	Medium
Web Developer	Medium

Most popular IT jobs in 2011



## Most popular IT jobs in 2012



## Most popular IT jobs in 2013



## Most popular IT jobs in 2014



## Most popular IT jobs in 2015





Most popular IT jobs in 2016



## Most popular IT jobs in 2017



## Most popular IT jobs in 2018



The above wordclouds generates the most popular jobs in a given year. Interestingly if we will not set a seed it will generate random wordclouds meaning each time running code we will get different wordclouds. Next we will analyze the required\_qualifications variable which represents the qualification and skills demanded by employee particularly in IT sector but code may easily be extended to other industries. First we analyze the whole dataset including all years from 2005 to 2018 then we may also conduct yearly analysis.

```
it_title_05_18 <- jobs_2005_2018[jobs_2005_2018$industry == "Information Technology",  
  ]
```

```
it_title_05_18_vs <- VectorSource(it_title_05_18$required_qualifications)
```

```
it_title_05_18_corpus <- VCorpus(it_title_05_18_vs)
```

```
it_title_05_18_dtm <- TermDocumentMatrix(it_title_05_18_corpus,  
  control = list(removeNumbers = T, stopwords = T, stemming = F,  
    removePunctuation = F))
```

```
it_title_05_18_dtm
```

```
## <<TermDocumentMatrix (terms: 32265, documents: 9457)>>
```

```
## Non-/sparse entries: 455936/304674169
```

```
## Sparsity : 100%
```

```
## Maximal term length: 79
```

```
## Weighting : term frequency (tf)
```

```
inspect(it_title_05_18_dtm)
```

```
## <<TermDocumentMatrix (terms: 32265, documents: 9457)>>
```

```
## Non-/sparse entries: 455936/304674169
## Sparsity      : 100%
## Maximal term length: 79
## Weighting      : term frequency (tf)
## Sample        :
##              Docs
## Terms      2369 3674 3997 4234 4417 6104 6306 6542 7211 9094
## ability    12   17   7   13   5   1   13   6   7   13
## english     0    0   0    0   1   1   1   2   2    0
## excellent   0    0   0    0   1   3   0   1   1   1
## experience   0    0   1    0   6   0   1   5   3   3
## good         0    0   0    0   0   1   0   0   1   3
## knowledge    9    5   5    9   0  18   1   0   3   5
## least        0    0   2    0   0   1   0   0   1   0
## skills;       0    0   0    0   1   2   0   2   0   3
## work         0    0   2    0   1   2   4   1   5   5
## years        0    0   2    0   1   1   0   0   1   1
```

```
dtm_mat <- as.matrix(it_title_05_18_dtm)

freqs <- rowSums(dtm_mat)

df_freq <- data.frame(terms = rownames(dtm_mat), freq = freqs,
  stringsAsFactors = F)

df_freq <- df_freq[order(df_freq$freq, decreasing = T), ]

head(df_freq, n = 40)
```

```
##              terms  freq
## knowledge      knowledge 20976
## experience     experience 19411
## ability        ability 12063
## work           work 9654
## skills;        skills; 8095
## good           good 8024
## years          years 6412
## excellent      excellent 6209
## english        english 6173
## least          least 5262
## strong         strong 5214
## communication  communication 4376
## degree         degree 4305
## skills         skills 4033
## computer       computer 3789
## procedures:    procedures: 3680
## plus;         plus; 3488
## development    development 3211
## russian        russian 2806
## understanding  understanding 2647
## working        working 2637
## software       software 2540
## related        related 2519
## team           team 2428
## university     university 2262
```

```
set.seed(42)
wordcloud(words = df_freq$terms, freq = df_freq$freq, min.freq = 20,
  max.words = 500, random.order = F, scale = c(2.5, 0.3), colors = brewer.pal(10,
    "Spectral"))
```



29

```
head(df_freq, n = 50)
```

```
##               terms  freq
## knowledge      knowledge 20976
## experience      experience 19411
## ability         ability 12063
## work            work    9654
## skills;         skills;  8095
## good            good    8024
## years           years    6412
## excellent       excellent 6209
## english         english  6173
## least           least    5262
## strong          strong    5214
## communication communication 4376
## degree          degree    4305
## skills          skills    4033
## computer        computer  3789
## procedures:     procedures: 3680
## plus;           plus;    3488
## development     development 3211
## russian         russian   2806
## understanding understanding 2647
## working         working   2637
## software        software  2540
## related         related   2519
## team           team     2428
## university      university 2262
## design          design    2249
## web            web      2226
## armenian        armenian  2115
## written         written   2093
## technical       technical  2063
## language        language  2051
## field;          field;    1990
## management      management 1924
## language;       language; 1904
## languages;      languages; 1855
## relevant        relevant   1772
## experience;     experience; 1691
## familiarity     familiarity 1645
## sql            sql       1627
## analytical      analytical 1615
## higher          higher    1609
## fluency         fluency   1520
## programming     programming 1416
## advanced        advanced   1404
## professional    professional 1394
## development;    development; 1311
## high           high      1238
## year           year      1180
## business        business   1139
## armenian,       armenian,  1124
```

As we see also from above table general words extremelly contaminate the results and don't allow us to see the technical skills needed which are demanded more in IT sector. That why in the following steps we will develop algorithm which will detect useful for analysis technical patterns such as a programming language and other technical skills to have clearer vision of the market and really insightfull results.

```
str(jobs_2005_2018)
```

```
## 'data.frame':   24288 obs. of  11 variables:
## $ company      : chr  "M-possible" "Lycos Europe" "Lycos Europe" "Lycos Europe" ...
## $ title        : chr  "Senior Software Engineer" "Technical System Documenter" "Flash A
## $ term         : chr  NA NA NA NA ...
## $ duration     : chr  NA "Permanent" "Permanent" "Permanent" ...
## $ location     : chr  "Yerevan, Armenia" "Yerevan, Armenia" "Yerevan, Armenia" "Yerevan
## $ job_responsibilities : chr  "Actively identify and implement tools, resources, new technologi
## $ required_qualifications : chr  "Minimum 3 years of experience in game industry; MS in Computer S
## $ opening_date  : Date, format: "2004-12-23" "2004-12-23" ...
## $ application_deadline_date: Date, format: "2005-01-31" "2005-01-15" ...
## $ job_id       : int  1699 1700 1701 1702 1703 1704 1706 1707 1708 1709 ...
## $ industry     : chr  "Information Technology" "Information Technology" "Information Te

programming_skills <- list("HTML", "CSS", "R", "PYTHON", "C",
  "C++", "C#", ".NET", "JAVASCRIPT", "JAVA", "SQL", "MYSQL",
  "SQL SERVER", "PHP", "JSP", "ASP", "UNIX", "ORACLE", "XML",
  "XSLT", "OOP", "OOD", "DHMTL", "FLASH", "APACHE", "ASP.NET",
  "LINUX", "APACHE", "MS ACESS", "WINDOWS")

skills_v <- c("HTML", "CSS", "\\s+R{1}\\s+", "PYTHON", "C\\+{1,2}",
  "C\\#", "\\s+.NET", "VISUALBASIC\\.NET", "JQUERY", "AGILE",
  "ASP\\.NET", "JAVASCRIPT", "JAVA", "SQL", "MYSQL", "SQL SERVER",
  "PHP", "JSP", "ASP", "UNIX", "ORACLE", "XML", "XSLT", "\\s+OOP\\s+",
  "\\s+OOD\\s+", "DHMTL", "FLASH", "APACHE", "ASP.NET", "LINUX",
  "APACHE", "MS ACESS", "WINDOWS")
```

Here we develop function to do wordclouiding for required\_qualifications variable to define the most demanded skills in IT on a yealry basis.

```
skills_f <- function(year = 2018) {
  it_df <- jobs_2005_2018 %>% mutate(Year = format(jobs_2005_2018$opening_date,
    "%Y")) %>% filter(Year %in% c(year) & industry %in% c("Information Technology"))

  skills_v <- c("HTML", "CSS", "\\s+R{1}\\s+", "PYTHON", "C\\+{1,2}",
    "C\\#", "GIT|GITHUB", "\\s+.NET", "VISUALBASIC\\.NET",
    "JQUERY", "AGILE", "ASP\\.NET", "JAVASCRIPT", "JAVA",
    "SQL", "MYSQL", "SQL SERVER", "PHP", "JSP", "ASP", "UNIX",
    "ORACLE", "XML", "XSLT", "\\s+OOP\\s+", "\\s+OOD\\s+",
    "DHMTL", "FLASH", "APACHE", "ASP.NET", "LINUX", "APACHE",
    "MS ACESS", "WINDOWS")

  skills_l <- str_extract_all(toupper(it_df$required_qualifications),
    pattern = paste(skills_v, collapse = "|"))
  skills_l <- na.omit(unlist(skills_l))
  skills_l <- data.frame(skills_l)

  skills_df <- skills_l %>% group_by(skills_l) %>% summarise(count = n()) %>%
    mutate(per = count/sum(count)) %>% arrange(desc(per))
```

```

layout(matrix(c(1, 2), nrow = 2), heights = c(1, 4))
par(mar = rep(0, 4))
plot.new()
text(x = 0.5, y = 0.5, labels = paste("Most demanded IT skills in ",
  year))
w <- wordcloud(words = skills_df$skills_1, freq = skills_df$count,
  min.freq = 50, scale = c(2.5, 0.3), random.order = F,
  colors = brewer.pal(8, "Dark2"), main = "Title")
}

for (i in 2005:2018) {
  skills_f(i)
}

```

Most demanded IT skills in 2005



C++  
LINUX SQL  
JAVA  
HTML  
WINDOWS



Most demanded IT skills in 2006



A word cloud representing the most demanded IT skills in 2006. The words are arranged in a triangular shape pointing upwards. The largest word is 'SQL' in the center. Other words include 'XML' (vertical on the left), 'JAVA' (top right), 'C++' (top center), 'C#' (top right), '.NET' (bottom left), 'HTML' (bottom right), and 'WINDOWS' (bottom center). The colors of the words are: XML (purple), JAVA (purple), C++ (green), C# (blue), SQL (black), .NET (pink), HTML (blue), and WINDOWS (pink).

Most demanded IT skills in 2007



Most demanded IT skills in 2008



Most demanded IT skills in 2009

SQL  
LINUX  
WINDOWS

Most demanded IT skills in 2010



Most demanded IT skills in 2011



Most demanded IT skills in 2012



Most demanded IT skills in 2013





Most demanded IT skills in 2014



Most demanded IT skills in 2015



Most demanded IT skills in 2016



Most demanded IT skills in 2017



## Most demanded IT skills in 2018



As we may clearly observe knowledge of SQL,HTML,JAVASCRIPT is highly demanded and appears practically in all wordclouds. As a matter of fact based on the wordclouds as times goes by more robust skillset is being demanded by employer from the applicants. This is quite natural as technology is developing and business owners want to see smart workers with vast amount of different specifications as no one knows exactly how technology would develop in the next few years so when employees pose different skillset employers minimize the risks for their business. We may also observe that role of GIT also increase over time particularly in 2017-2018 years. As a matter of fact it hard to grasp all aspects of wordclouds and conclusions will be highly subjective depending on the user so we will overcomplicate the interpretations based on the wordclouds and will leave it on the users of this analysis.

To conclude, in our analysis we scraped data from an Armenian online job postings website for period from 2005 to 2018 year and conducted analysis aimed at identifying the patterns in Armenian job market. We looked at historical rates of online job postings and revealed that the number of online job postings significantly increased and the majority of them is in IT sector. We emphasized this job industry in our analysis but we may easily extend in to any predefined job category defined in our dataset. We revealed the most popular job titles i.e. professions in Armenia which are presented year by year using wordclouds. In addition, we discovered the programming languages or skills that are highly demanded by employers also on yearly basis from 2005 to 2018. Ultimately there is much more to discover in the data but for the purpose of this analysis we will not go further and may continue in the future projects.