The Starving Software Tester: An Attempt at Observing Trends in Software Testing Recruitment

[1]

Number of words: ~ 2193 (excluding titles, references and figures).

Date of submission of the assignment: 21/10/2025

Anita Pal / ap224pu@student.lnu.se

[1] https://www.freepik.com/premium-vector/software-testing-found-bug-your-web-site-have-problem_31705233.htm

Introduction

Luciano Baresi and Mauro Pezzè (2006) state that software testing is integral to the software development process, covering the entire journey from requirements to the post-deployment stage, including activities such as maintenance and post-mortem analysis. It involves assessing whether a product is ready for release, while also checking for any software flaws (p. 90). With the rise of AI, the landscape of software testing is evolving, although the adoption of AI practices has been slower than in other engineering areas (Karhu et al., 2025, p. 1).

The following digital humanities mini-project examines job descriptions[2], which form the material component (cf. Drucker, 2021, p. 2), using various visualisations from an employment website to depict the presentation component (cf. Drucker, 2021, p. 1). For the processing part of the project (cf. Drucker, 2021, p. 1), the project makes use of web scraping—a technique that Ryan Mitchell (2018) defines as involving collecting data through an automated script that queries a server, requests data, and then parses the data to receive the requested information (p. x). Keeping ethics in mind, the data gathered from the job board is publicly accessible and not traceable to any individuals (Markham & Buchanan, 2017, pp. 203 - 204).

Further, web scraping refers to activities that can capture information from pages that humans can read (Black, 2016, p. 95), but – unlike data scraping, as Shehu Mustapha et al (2024) state – it only extracts information that is available online (p. 294). Most importantly, web scraping helps to convert unstructured data into a more structured format for further analysis (p. 290).

(262 words).

Visualisations in DH

As Edward Vanhoutte (2016) remarks, Digital Humanities (DH) began developing after World War II, when computing resources were no longer needed solely for the war effort; however, its roots trace back much earlier, to Ada Lovelace and her work on Babbage's Analytical Engine. For Ada, the work was not just restricted to the calculations of numbers, but could be engaged in other areas such as music (pp. 121-122).

Nowadays, DH, as defined by Johanna Drucker (2021) is viewed as a marriage of humanistic materials with digital methods (p. 1), that, according to Martin Paul Eve (2022), is sometimes perceived as a threat to the traditional humanities, not only stripping those fields of their much-needed funding but also threatening the foundations of humanities by perverting critical thinking with technological or digitial solutions (p. 1).

However, Eve (2022) highlights that DH — rather than removing the scholar from literary texts, for example — brings them closer to the material, as the application of digital methods and activities, such as mapping or building databases, allows for a new manner of engagement (p. 4), while also allowing to answer questions such as Stephen King's usage of adverbs in a much faster and efficient way than reading his novels in order and making notes of how adverbs show up in his various novels (p. 8). Beyond that, scholars such as Anne Burdrick et al. (2012) perceive DH as an opportunity to increase collaboration with other academic fields, thus not just expanding the area of humanistic studies but also permitting future researchers to build upon these new developments and deepen their research (p. 3).

[2] https://testdevjobs.com/location/remote-united-kingdom

Drucker (2014) states that data, referred to as "*capta*", which humans use to understand and familiarise themself with the world surrounding them, must be fabricated or manufactured (p. 128). It stands in contrast, Drucker (2021) explains, with the term "*data*", which some perceive as easily accessible and ready for a lucky data hunter to come across without any need for further modifications (p. 2). The term "*capta*" in relation to the humanities stems from the fact that scholars in that field approach an enquiry from a more evaluative stance, rather than simply measuring or observing phenomena as in the sciences (Masson, 2017, p. 25).

Supporting this idea, Shawn Day (2022) suggests that the humanities prefer exploration and discovery to systematically arranging things and accepting the boundaries presented by them (p. 211), even though, as Miriam. Posner (2015) describes that the data used in DH is not usually created via experimentation or observation, but mined from existing historical sources (p. 1).

Furthermore, Day (2022) emphasises that the humanities tend to view representations of data critically, more likely to step back and consider the data processing involved before jumping to any conclusions (p. 212). For Day, the purpose of visualisation in the humanities is to bring people together by provoking them in the hopes of encouraging more discussion on a given subject matter (p. 214). Posner (2015) clarifies that this is because the data in the DH is often complex, requiring solutions that require creativity and even new data specifications when existing standards do not meet the needs of a DH-specific project (p. 1).

Visualisations created by digital tools can make data more easily processable and enthralling to the public in a field like cultural heritage (Münster et al, 2019, p. 814), because of how visual material serves as its own rhetoric that captures complex data better than other methods of depiction (Drucker, 2021, p. 86). For example, in the case of the networks that, despite primarily serving as mathematical tools, can also entice a casual viewer to see things in novel ways, often serving as a narrative (T. Venturini et al, 2017, pp. 155-156) that produces meaning (Drucker, 2021, p. 86).

In this manner, visualisations mimic the power of databases, which allow vast amounts of data to be processed and displayed. These visualisations can provide the starting point for a discussion of the presented data (Hayes, 2007, pp. 1603-1605). Additionally, Maureen Engel (2018) asserts that maps, synonymous with the rapid development of technology, are integral to people's everyday lives, inhabiting the space between humans and the places they live in. These spaces tell stories, weaving tales of complexly spun relationships (pp. 212, 217).

One of the most significant risks in DH, Mark Hull (2020) argues, is that researchers, often intimidated by the bulk and volume of data available to analyse, fail to consider whether everything in that dataset is actually worthy of being "fed into an [...] algorithm" (p. 57). Therefore, the purpose of this mini-project is to encourage discussion on how the domain of software testing is evolving and to highlight how poorly processed data (Hall, 2020, p. 56) can lead to incorrect visualisations, necessitating a step back before making any further judgements.

Another risk in the DH, says Qiuzi Guo (2024), is the rise of AI, which allows humanists with no coding experience to apply computational methods to DH projects with the use of ChatGPT. The risk is that LLMs (Large Language Models) may be able to process a vast quantity of data. Still, they cannot fine-tune or access the depth of the material, opting for patterns over the nuances and complexities the data may present. Therefore, Guo suggests that tools like ChatGPT should be used to help answer creative questions, rather than replace the need for programming (pp. 59-60, 76).

(900 words).

Project Discussion

The purpose of this mini-project is to encourage discussion on how the domain of software testing is evolving. It also aims to highlight how poorly processed data (Hall, 2020, p. 56) can lead to incorrect visualisations, necessitating a bit more pondering before making any further judgements.

The functions in Figure 1 process the scraped data, with *traverseJobs* serving the purpose of getting the job listing from the URL to be collected (via calling the *getJobListings* function) and, once the data has been gathered, iterating through each job in the found listings to extract the required information and append them to empty lists; these empty lists are ultimately merged into a data frame[3].

Figure 1
The functions involved in scraping the data from the employment site.
Note. Created in Google Colab.

```
def getJobListings(driver: str, url: str):
  driver.get(url)
  job_listings = _driver.find_elements(By.XPATH, "//*[@class='job-deatils']")
  return job_listings

def fetchJobDetailsAndAddToList(driver: str, selector: str, empty_list):
  job_detail = driver.find_element(By.XPATH, selector).text
  empty_list.append(job_detail)

def traverseJobs(driver: str, url: str):
  job_listings = getJobListings(driver, url)
  job_count = 0
  job_title = []

  job_location = []
  job_skills = []
  job_id = []

  for job in job_listings:
    fetchJobDetailsAndAddToList(driver, "//*[@class='ml-5 jobtitle is-size-5 has-text-weight-semibold']", job_title)
    fetchJobDetailsAndAddToList(driver, "//*[@class='tags ml-5 mr-1 mb-0']", job_location)
    fetchJobDetailsAndAddToList(driver, "//*[@class='tags ml-5 mr-1']", job_skills)
    job_count += 1
    job_id.append(job_count)

  jobs_panda_frame = pd.DataFrame(list(zip(job_id, job_title, job_location, job_skills)), columns = ['ID', 'Title', 'Location', 'Skills'])
  return jobs_panda_frame.to_csv('jobs.csv')

traverseJobs(_driver, _url)
```

To prepare the data for Voyant[4], the first step involved further modifying the frame by restricting its columns to 'Skills', which portray the required qualifications in the previously scraped job descriptions (Figure 2). This modified data frame was then saved as a separate CSV file, titled 'skills.csv'. The created diagram – shown in Figure 3 – would make a layperson believe that, in the job descriptions provided by the data so far, all the skills - such as Python and Pytest – are equally desirable, suggesting that this is where the future of software testing lies; however, a look at the CSV file from the initial web scraping shows that the data had been duplicated.

Figure 2
The code used to further filter the data based on skills.
Note. Created in Google Colab.

```
# Prepare data for Voyant processing

jobs_data = pd.read_csv("/content/jobs.csv")
skills_data = jobs_data["Skills"]
skills_data.head()

# Save into a new CSV file
skills_data.to_csv("skills.csv")
```

Figure 3

[3] https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html
[4] https://voyant-tools.org/docs/tutorial-cirrus.html

The result of loading the aforementioned ‘skills.csv’ file into Voyant.
Note. Screenshot of graph created in Voyant.

An alternative to using Voyant was available by employing a word cloud in Python[5], which allowed for more customisation, as shown in Figure 4. In this case, the CSV data frame containing the jobs was not restricted by columns because this was handled during the creation of the word cloud graph (i.e. not requiring the creation of a separate CSV file).

Figure 4
The code below is responsible for creating a word cloud.
Note. Created in Google Colab.

```
# Create a word cloud
text = " ".join(skills for skills in jobs_data.Skills.astype(str))
stopwords = set(STOPWORDS)
wordcloud = WordCloud().generate(text)

# Customise the word cloud
wordcloud = WordCloud(width=800, height=400, stopwords = stopwords, background_color='lightblue', max_words=100, colormap='Purples_r').generate(text) #
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('on')
plt.title("Desired Software Testing Skills")
plt.show()
```

Nevertheless, while the resulting graph (Figure 5) is arguably beautiful to look at, it would —if ever made public —paint a very wrong picture of what employers would want from prospective software testers, with terms such as Azure and GCP displayed prominently. The mistakes in this graph illustrate how data is something human-made (cf. Drucker, 2014, p. 128) and emphasise that the application of tools must be undertaken carefully (cf. Hull, 2020, p. 56). Despite that, the visual power of the graph cannot be underestimated (cf. Drucker, 2021, p. 86) as its bright colours tell a story of Azure's incredible popularity, which demonstrates Microsoft's continuing dominance.

Figure 5
A word cloud generated in Python.
Note. Created in Google Colab.

The difficulty of using flawed data becomes even more prominent in the subsequent visualisation attempt, which concerned displaying the central job locations using Palladio's network feature. Firstly, to prepare the data for Palladio[6] processing, the data had to be cleaned using Python code, as shown in Figure 6:

[5] https://www.geeksforgeeks.org/python/generating-word-cloud-python
6 https://hdlab.stanford.edu/palladio-app/#/upload

Figure 6
Cleaning up data for Palladio processing.
Note. Created in Google Colab.

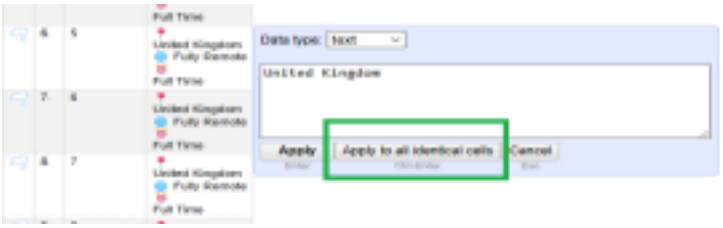
```
# Prepare data for Palladio processing before working with OpenRefine.

# Remove repetitions
jobs_data.drop_duplicates(inplace = True)
jobs_data.head()
jobs_data.to_csv("updated_jobs.csv")

location_data = jobs_data["Location"]
location_data.to_csv("locations.csv")
location_data.head()
```

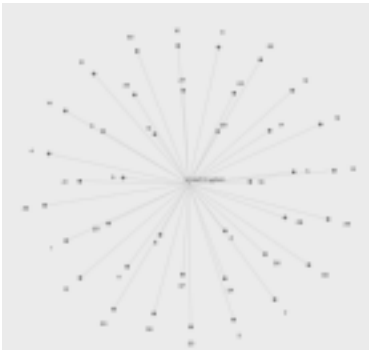
Afterwards, the data had to be further modified, and OpenRefine[7] proved effective in how changes applied to one column could be carried over to others (Figure 7). While in this scenario, the much-needed editing of data before visualisation proved inefficient, it does showcase how the processing stage can require multiple steps (cf. Drucker, 2021, p. 1). With the data modified, an appropriate CSV could be exported and -- thereupon -- uploaded to Palladio.

Figure 7
OpenRefine can be helpful for further data processing.
Note. Screenshot of data-processing action carried out in OpenRefine.



The network graph in Figure 8 suggests that the job locations from the scraped data are all found in the United Kingdom. Still, it does not provide any additional information, indicating that lacking or insufficient data can lead to equally uninformative graphs that neither provide a narrative nor engage the viewer in any specific way (cf. T. Venturini et al, 2017, pp. 155-156).

Figure 8
A network graph created in Palladio shows how the job locations are mainly in the United Kingdom.
Note. Screenshot of graph created in Palladio.



7 https://openrefine.org

As with Voyant previously, there was an alternative to Palladio that utilised the NetworkX[8] library, as shown in Figure 9, producing a graph (Figure 10) with a few more colours than the one created in Palladio. Still, while the graph is more visually interesting, its narrative does not convey much beyond 38 nodes connected to a single location, making it of little use for an in-depth discussion (cf. Hayes, 2007, pp. 1603-1605).

Figure 9
The code used to create a network graph in Python.
Note. Created in Google Colab.

```
# Create a network graph @ https://medium.com/@harshkjoya/connecting-the-dots-creating-network-graphs-from-pandas-dataframes-with-networkx-9c4fb60089cf
G = nx.Graph()
G.add_nodes_from(locations_csv['Column'])
G.add_nodes_from(locations_csv['Location'])
edges = [(row['Column'], row['Location']) for index, row in locations_csv.iterrows()]
G.add_edges_from(edges)

# Draw the graph
pos = nx.kamada_kawai_layout(G)
nx.draw(G, pos, with_labels=True, node_size=1500, node_color='purple', font_size=25, font_color='black', edge_color="pink")
plt.title("Locations")
plt.show()
```

Figure 10
A network graph generated in Python.
Note. Created in Google Colab.

The final round of experiments for this mini-project involved using Microsoft Copilot[10] to verify whether it understood human prompts (or whether my prompts were understandable) to generate DH-related visualisations. For instance, without the need for any data, asking about the most desirable skills for software testers in the UK prompted the following exchange and visualisation (Figure 11):

Figure 11
AI-generated word cloud.
Note. Screenshot of a graph created by Microsoft Copilot.

Although the LLM was lightning-fast and could create pretty visualisations on the fly, it did not always perform as expected, with my request for several visualisations related to the uploaded CSV file not showing up in the GUI, and even though it generated code to create the visualisations, modifications were required before being able to process it in Google Colab (Figure 12), showcasing

8 https://networkx.org

that AI will never quite replace the common sense that humans do occasionally display (cf. Guo, 2024, p. 76).

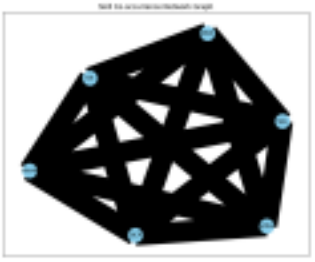
Figure 12
Snippet of the AI-generated code with modifications.
Note. Created by Microsoft Copilot.

```
# Load UK map # MODIFIED https://github.com/mattijn/topojson/issues/224
world = gpd.read_file(
    "https://d2ad6b4ur7yvpq.cloudfront.net/naturalearth-3.3.0/ne_50m_admin_0_countries.geojson"
)
uk = world[world.name == 'United Kingdom']

# Create dummy point for remote jobs
remote_point = gpd.GeoDataFrame([{'geometry': Point(-1.5, 54.0), 'Location': 'Remote'}], crs='EPSG:4326')
```

Lastly, the visualisations created by the AI -- apart from the network graph related to skills (Figure 13) -- did not offer any improvements over previous work, showing that caution and critical thinking are required when engaging with such tools (cf. Day, 2022, p. 212). The network graph stands out because it suggests that the skills needed in software testing are often co-dependent.

Figure 13
AI-generated network graph.
Note. Created in Google Colab.



(851 words).

Conclusion

In a nutshell, the following mini-project, drawing on research provided by other DH scholars, explores the opportunities of expanding humanistic research when working together with digital methods. It also reveals the need to maintain a critical stance when dealing with data, as visualisations or presentations can only be as beneficial as their underlying data. If the data has not been processed correctly, then the results will be lacklustre.

Consequently, for budding DH scholars, it is vital to use computational tools appropriately, taking nothing for granted while embracing the subject from a playful, innovative point of view. After all, DH is not just a field that allows for a new perspective of the humanities (cf. A. Burdrick et al, 2012, p.3), but one for the sciences, where simple observation and recording (cf. Masson, 2017, p. 25) give way to a new, more creative approach that emphasises collaboration, mutual support and a sense of adventurousness. As a closing remark, DH can be seen as an exciting field that has the potential to bring people together (cf. A. Burdrick et al, 2012, p.3).

(180 words).

Acknowledgements

The report did not utilise any AI tools, except for those used for visualisation and Grammarly, which was employed as a sanity check to identify any glaring errors.

References

Baresi, L., & Pezzè, M. (2006). An Introduction to Software Testing. *Electronic Notes in Theoretical Computer Science*, 148(1), 89–111. https://doi.org/10.1016/j.entcs.2005.12.014.

Burdick, A., Drucker, J., Lunenfeld, P., Presner, T., & Schnapp, J. (2012). *Digital Humanities*. MIT Press.

Day, S. (2023). Visualising humanities data. In J. O'Sullivan (Ed.), *The Bloomsbury Handbook to the Digital Humanities* (pp. 211-219). Bloomsbury Publishing.

Drucker, J. (2014). *Graphesis: visual forms of knowledge production*. Harvard University Press.

Han, J., Kamber, M., & Pei, J. (2012). Data mining: Concepts and techniques (3rd ed.). Morgan Kaufmann.

Drucker, J. (2021). *The digital humanities coursebook: An introduction to digital methods for research and scholarship* (1st ed.). Routledge. https://doi.org/10.4324/9781003106531

Engel, M. (2018). Deep Mapping: Space, Place, and Narrative as Urban Interface. In Sayers, J (Ed.), *The Routledge Companion to Media Studies and Digital Humanities* (pp. 214-221). Routledge. https://doi.org/10.4324/9781315730479

Eve, P.E. (2022). *The Digital Humanities and Literary Studies*. Oxford University Press. https://doi.org/10.1093/oso/9780198850489.001.0001

Hall, M. (2020). Opportunities and Risks in Digital Humanities Research. In H. Carius, M. Prell & R. Smolarski (Eds.), *Kooperationen in den digitalen Geisteswissenschaften gestalten* (pp. 47–66). Vandenhoeck & Ruprecht GmbH & Co.

Hayes, K. N. (2007). Narrative and Database: Natural Symbionts. *Special Topic: Remapping Genre*, 122(5), 1603-1608. https://www.jstor.org/stable/25501808

Guo, Q. (2024). Prompting Change: ChatGPT's Impact on Digital Humanities Pedagogy – A Case Study in Art History. *International Journal of Humanities and Arts Computing*, 18(1), 58-78. https://doi.org/10.3366/ijhac.2024.0321

Karhu, K., Kasurinen, J., & Smolander, K. (2025). Expectations vs Reality -- A Secondary Study on AI Adoption in Software Testing. *Software Testing, Verification and Reliability*, 1-26. https://doi.org/10.48550/arXiv.2504.04921

Klein, L. F. (2018). Timescape and Memory: Visualizing Big Data at the 9/11 Memorial Museum. In J. Sayers (Ed.), *The Routledge Companion to Media Studies and Digital Humanities* (pp. 433-444). Routledge.

Markham, A., & Buchanan, E. (2017). "Research Ethics in Context: Decision-Making in Digital Research. In K. van Es, & M. T. Schäfer (Eds.), *The Datafied*

Society: Studying Culture through Data (pp. 155-170). Amsterdam University Press. https://oapen.org/search?identifier=624771

Masson, E. (2017). Humanistic Data Research: An Encounter between Epistemic Traditions. In M. T. Schäfer & K.v. Es (Eds.), *The Datafied Society. Studying Culture through Data* (pp. 25-37). Amsterdam University Press.

Mitchell, R. (2018). Web Scraping with Python (2nd ed.). O′Reilly. https://www.oreilly.com/library/view/web-scraping-with/9781491985564/

Münster, S., Apollonio, F. I., Bell, P., Kuroczynski, P., Di Lenardo, I., Rinaudo, F., & Tamborrino, R. (2019). DIGITAL CULTURAL HERITAGE MEETS DIGITAL HUMANITIES. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 813–820. https://doi.org/10.5194/isprs-archives-XLII-2-W15-813-2019

Mustapha, S., Man, M., Bakar, W. A. W. A., Yusof, M. K., & Sabri, A. A. (2024). Demystified Overview of Data Scraping. *International Journal of Data Science and Advanced Analytics*, 6(6), 290-296. 10.69511/ijdsaa.v6i6.205.

Posner, M. (2015, June 25). *Humanities Data: A Necessary Contradiction* [Blog post]. Miriam Posner's Blog. http://miriamposner.com/blog/humanities-data-a-necessary-contradiction/.

Vanhoutte, E. (2016). The Gates of Hell: History and Definition of Digital | Humanities | Computing. In M. Terras, J. Nyhan & E. Vanhoutte(Eds.), *Defining Digital Humanities: A Reader* (pp. 119-156). Routledge. https://doi.org/10.4324/9781315576251

Venturini, T., Bounegru, L., Jacomy, M., & Gray, J. (2017). How to Tell Stories with Networks: Exploring the Narrative Affordances of Graphs with the Iliad. In K. van Es, & M. T. Schäfer (Eds.), *The Datafied Society: Studying Culture through Data* (pp. 155-170). Amsterdam University Press. https://oapen.org/search?identifier=624771