Navigating Work Skill Readiness Using ChatGPT

Thomas McCullough and Ram Dantu

10th Annual CAE in Cybersecurity Community Symposium - Cyber Defense Track Proposals

ACKNOWLEDGEMENTS: We sincerely acknowledge and thank the National Centers of Academic Excellence in Cybersecurity, housed in the Division of Cybersecurity Education, Innovation and Outreach, at the National Security Agency (NSA) for partially supporting our research through grants H98230-20-1-0329, H98230-20-1-0414, H98230-21-1-0262, H98230-21-1-0262, and H98230-22-1-0329.

About Us:

Thomas McCullough
Ph.D. Student
Department of Computer Science and Engineering
University of North Texas, Denton, Texas, 76203
Awarded for Outstanding Undergraduate Work at UNT - Department of Computer Science and Engineering
Contact: thomasmccullough@my.unt.edu
www.github.com/Crimso777

Dr. Ram Dantu
Professor
Department of Computer Science and Engineering
Director, Center for Information and Cyber Security (designated for academic excellence by NSA and DHS, cics.unt.edu)
University of North Texas, Denton, Texas, 76203.
www.cse.unt.edu/~rdantu

Our work thus far:

- Working to match job applicants to optimal roles in industry
- Relied on rigid NLP framework for skill extraction
- Very promising results but required lots of manually labeled data
- Encouraged by results, we asked: “How could we make this process better?”

Leveraging Large Language Models

- **Increased accuracy**: LLMs are trained on massive amounts of data and can make connections that traditional techniques may miss
- **Flexibility and Adaptability**: LLMs are effective across different inputs and contexts. This means less manual tweaking between different courses and job postings
- **Faster development time**: The hardest part of implementation an LLM is the training process. Since this is already done for many models, we can save development time
- **Scalability**: LLMs are built to handle massive amounts of data, thus we can easily increase the throughput of the pipeline without major changes

Motivation

- In our results thus far, Chat-GPT generates far more soft skills than our more rigid models
- Chat GPT isn’t limited by a list of approved skills and can generalize similar skills automatically, while still maintaining diversity
- Major changes to the pipeline can be done quickly to include different categories of skills
- Many techniques in current literature for optimizing LLM

Why Chat-GPT?

- Outperforms other similar models
- New industry standard for variety of NLP tasks
- Highly customizable between prompt engineering and fine tuning
- Cheap and scalable

Differences between ChatGPT and Traditional Language Models

Features	ChatGPT	Traditional Algorithms
Corpus Size for Training	Over 45 terabytes of text	Several Gigabytes to a few terabytes of text
Fine tuned capability	Specific NLP Tasks using specialized datasets	Can also be fine-tuned for specific use cases, but requires additional training data and resources
Personalization	Generates responses based on the context	Pre-programmed with a limited set of responses
Model size	Over 175 billion parameters	Over 110 million to 340 million parameters
Pre-training	Pre-trained on a larger text data	Pre-trained on smaller text data
Use Cases	More adaptable for unknown domains, hence flexible	Specific domain and poor adaptability, hence brittle

Why not build our own?

- Over $4 million to build
- Over $700,000 per day to maintain
- Millions of GPU hours to train
- Training requires human feedback and is sensitive to inputs (we could mess it up)

The Problem

- Recent computer science graduates have a 7.8% unemployment rate, over twice the national average
- Number of Computer Science and IT positions will grow by 12% over the next 10 years.
- Despite many candidates, new hires are missing fundamental skills
- Disparity between the candidate we want and the candidate we get
- The right candidate should be out there, but we cannot find them

https://www.synergisticit.com/tech-companies-not-hire-computer-science-graduates/

Our Solution

- Create a micro-credits of a graduate’s skills
- Weight skills based on grades
- Generate a list of skill requirements based on a job posting
- Match the perfect candidate to their industries requirements

Chat-GPT Basics

- Prompts are a string of natural language fed into the model
- String length is limited based on tokens (4096 for Chat-GPT)
- Tokens are atomic Natural Language units, usually a word or part of a word
- Techniques can be applied to select a prompt with highest accuracy

Chat-GPT Strategies

- Prompt engineering refers to any automatic modification to a hand-generated prompt
  - Few-Shot Examples are sample input-output sequences fed to Chat-GPT to increase accuracy
  - Chain of Thought prompting instructs Chat-GPT to emulate a sequence of rational reasoning steps when presenting results. This can increase accuracy even when no additional steps are required
- Fine tuning can be used to tweak the model itself to be more effective at a particular task
  - Expensive
  - Very sensitive to biases in data
- Manual adjustment of token probabilities
  - Can be used to remove bias from a prompt directly

Assessments Step 1 (Scraping)

- A Natural Language pipeline needs raw text, and existing strategies are sufficient for our purpose
- Many assessments available online
- **Input**: a CAE Institution website with Assessments posted
- **Output**: raw text of assessments and metadata in a *pandas* dataframe

Assessments Step 2 (Segmentation)

- Proper segmentation of assessments is key to proper skill extraction
- Segmentation necessary to take advantage of prompt engineering, because token limits restrict the number of examples
- Segmentation based on Instructions, code segments, and questions
- Misclassification can lead to wildly different results(computer security course considered an artificial intelligence course if python code the only thing in a segment labeled question)

Assessment Step 3 (Prompt Engineering)

- Calculate optimal set of examples for a given task, for instance code segments require different prompt than tutorial segments
- Manually label examples or automatically evaluate accuracy through mutual information
- Include labeled few-shot examples in prompt for skill extraction
- Apply rationale Chain-of-Thought prompting for added accuracy

Step 4 (Job skill extraction)

- Job postings much shorter than assessments, which is significant when considering token limit
- Few shot possible without segmentation
- Skills usually higher level but also explicitly stated
- Distinct prompt required that is different from assessments prompts

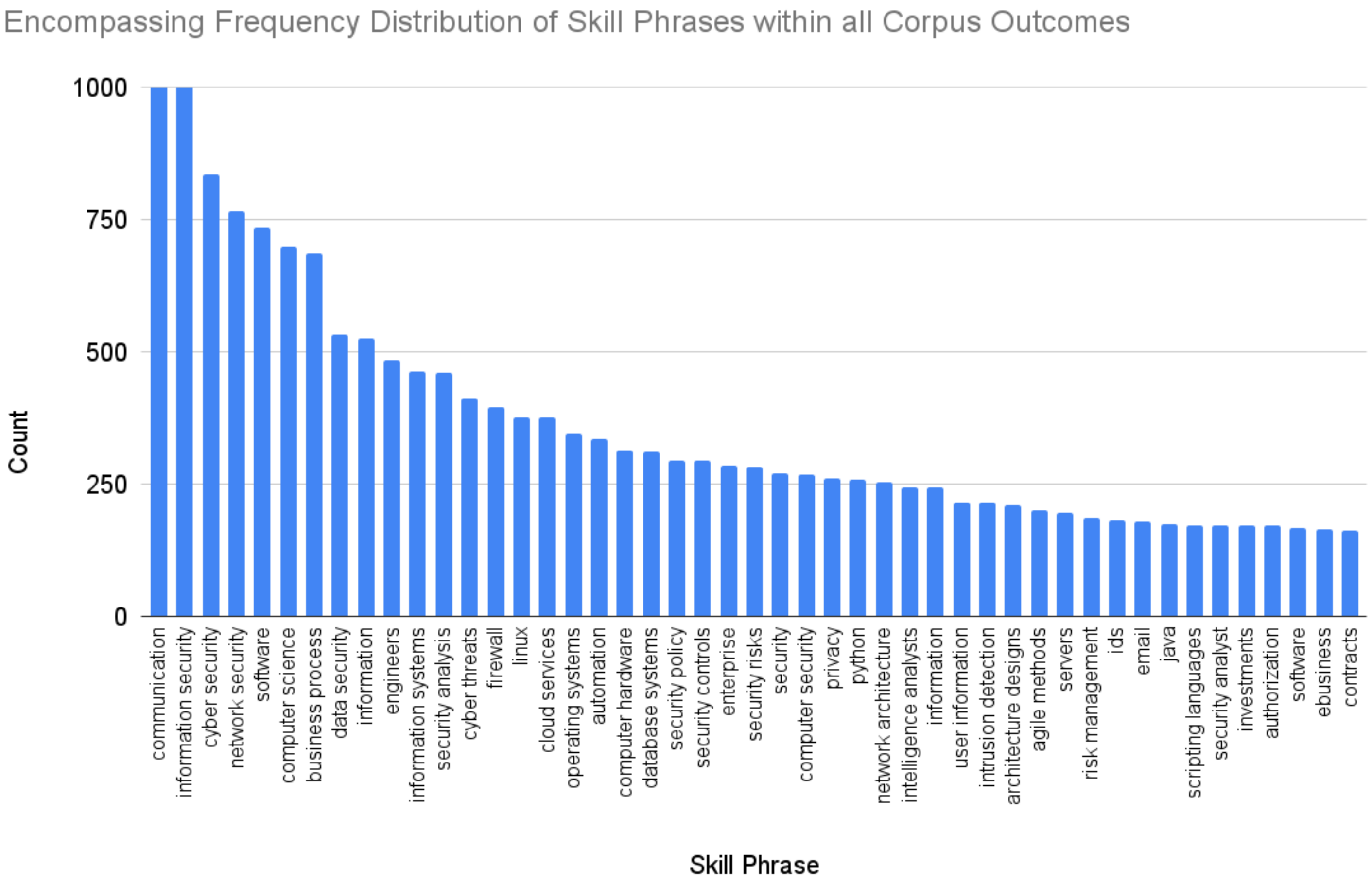
Step 5 (Comparisons)

- To be implemented in the future, but essential to pipeline
- Compare cosine similarity between extracted skills from each group
- Extract most compatible skills and modulate compatibility score by grades received
- Possible for employers to query specific skills from a set of students and vice versa

Methodology

- We hand-crafted 4 generations of prompts, iterating on the previous version for each iteration
- Skills extracted via sessions with Chat-GPT API from real job postings scraped from Indeed.com
- Accurately labeled skills defined as a manually labeled skill with a corresponding Chat-GPT generated skill with at least .7 cosine similarity using Spacy large English corpus
- For segmentation, each segment is a subset of the job posting where the integrity of sentence structure is maintained
- We did not use any additional selection process as to demonstrate the dangers of arbitrary segmentation with respect to accuracy
- We tested with 4 levels of segmentation where on average there were 1 segment, 1.5 segments, 2.0 segments, and 4.0 segments given a single job posting

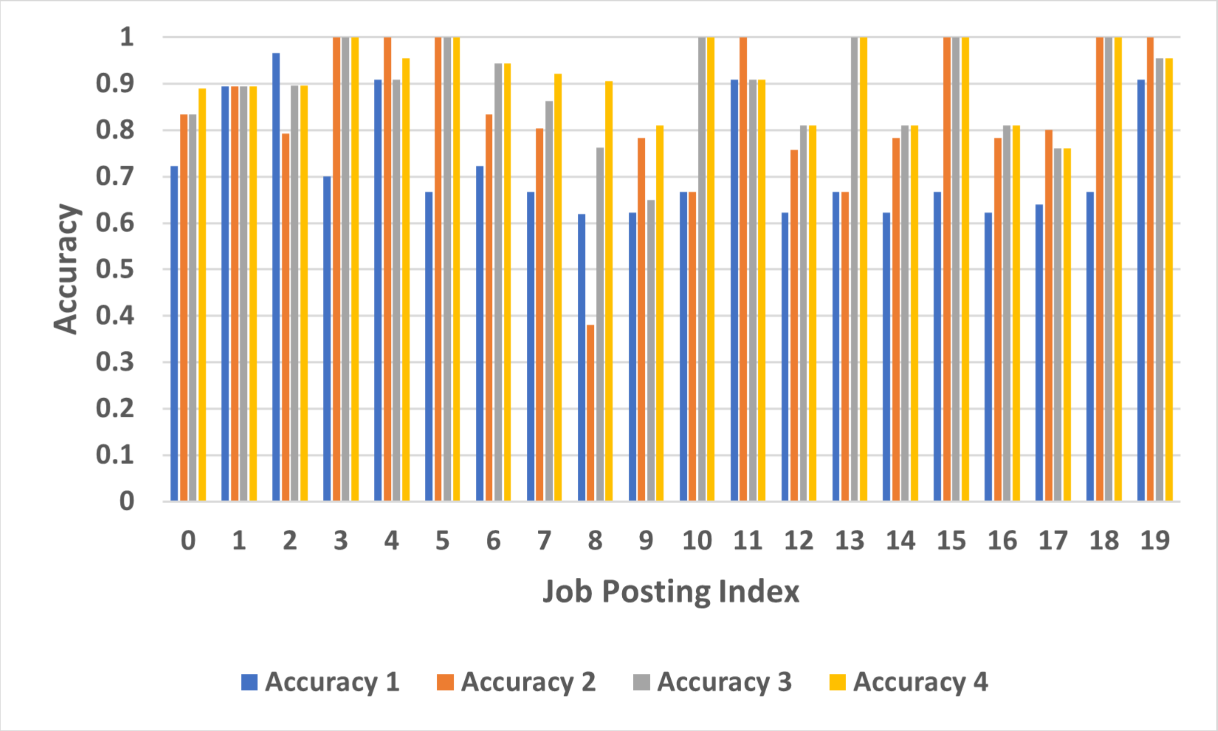
Frequency Distribution of the Skill Phrases

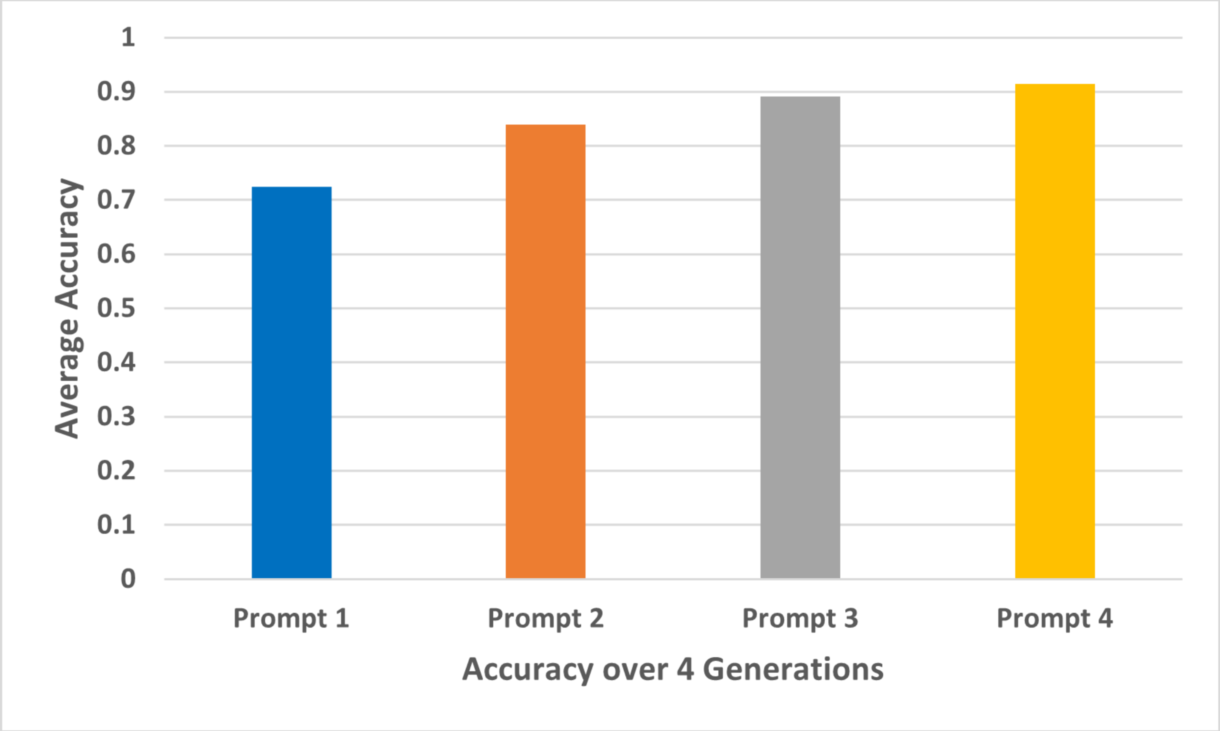


<u>Insights</u>:

- A frequency distribution of the 50 most frequent skill phrases is plotted against that phrase's count
- Some phrases seem trivially expected including, *cyber security* and *computer science*
- However, other phrases including, *linux, security analytics, network security,* and *data security* make the entire data set seem biased towards a specific area of computer science
- Most notably, the phrases *communication* and *information security* were counted just over 1,000 different times across all samples.

Total Accuracy over 4 Generations of hand-crafted prompts



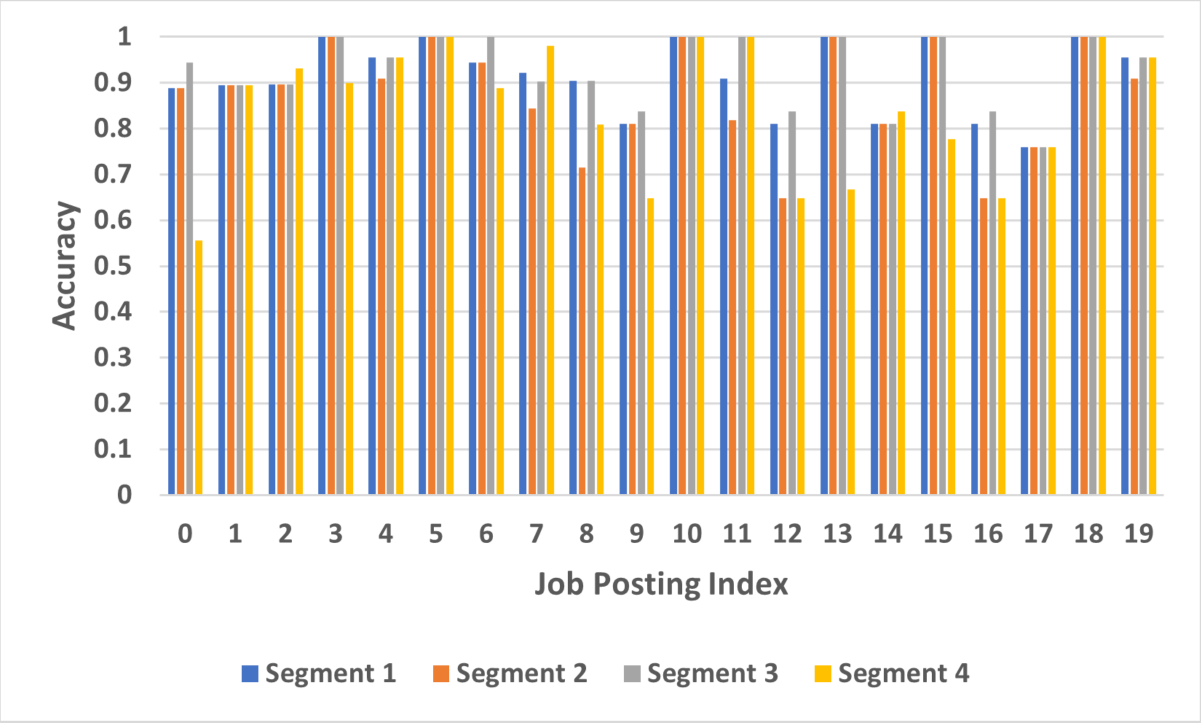


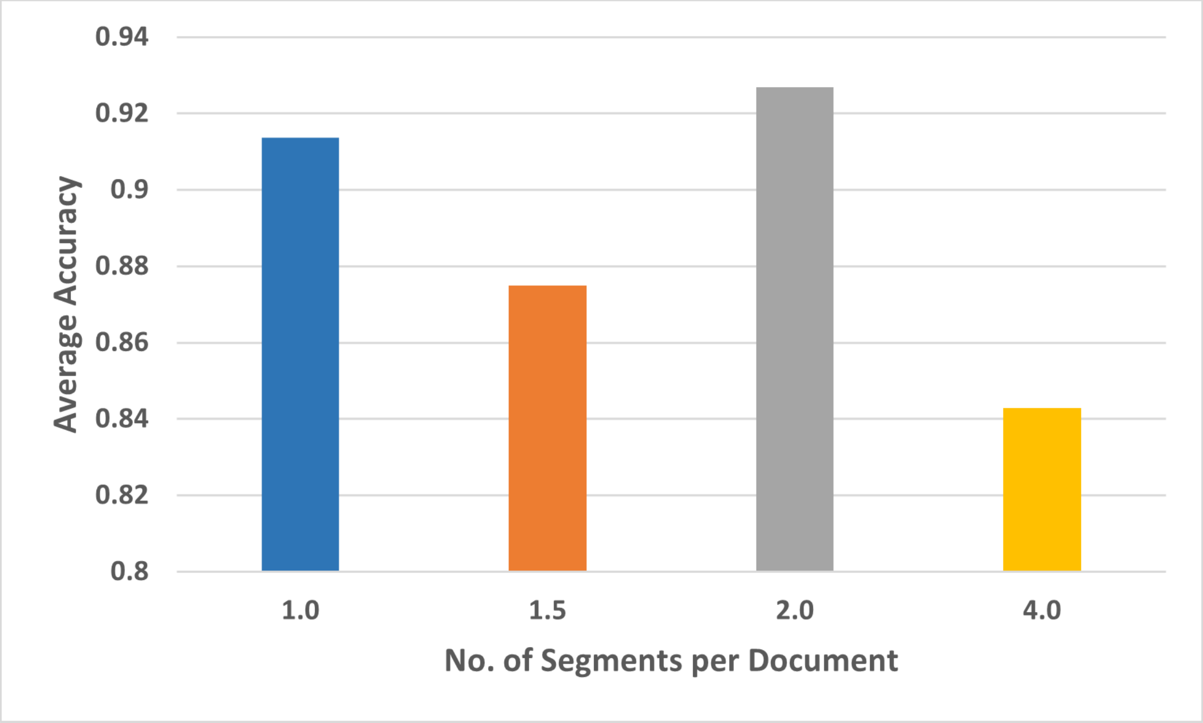
- After each result, prompt manually adjusted to better extract skills

Insights:
- Skill extraction heavily dependent on prompt quality
- Prompt engineering is non-trivial and critical to results
- Few-Shot examples can dramatically increase accuracy according to Liu et al
- Segmentation necessary to include more examples

https://arxiv.org/abs/2101.06804

Accuracy given 4 arbitrary Segmentation Methods



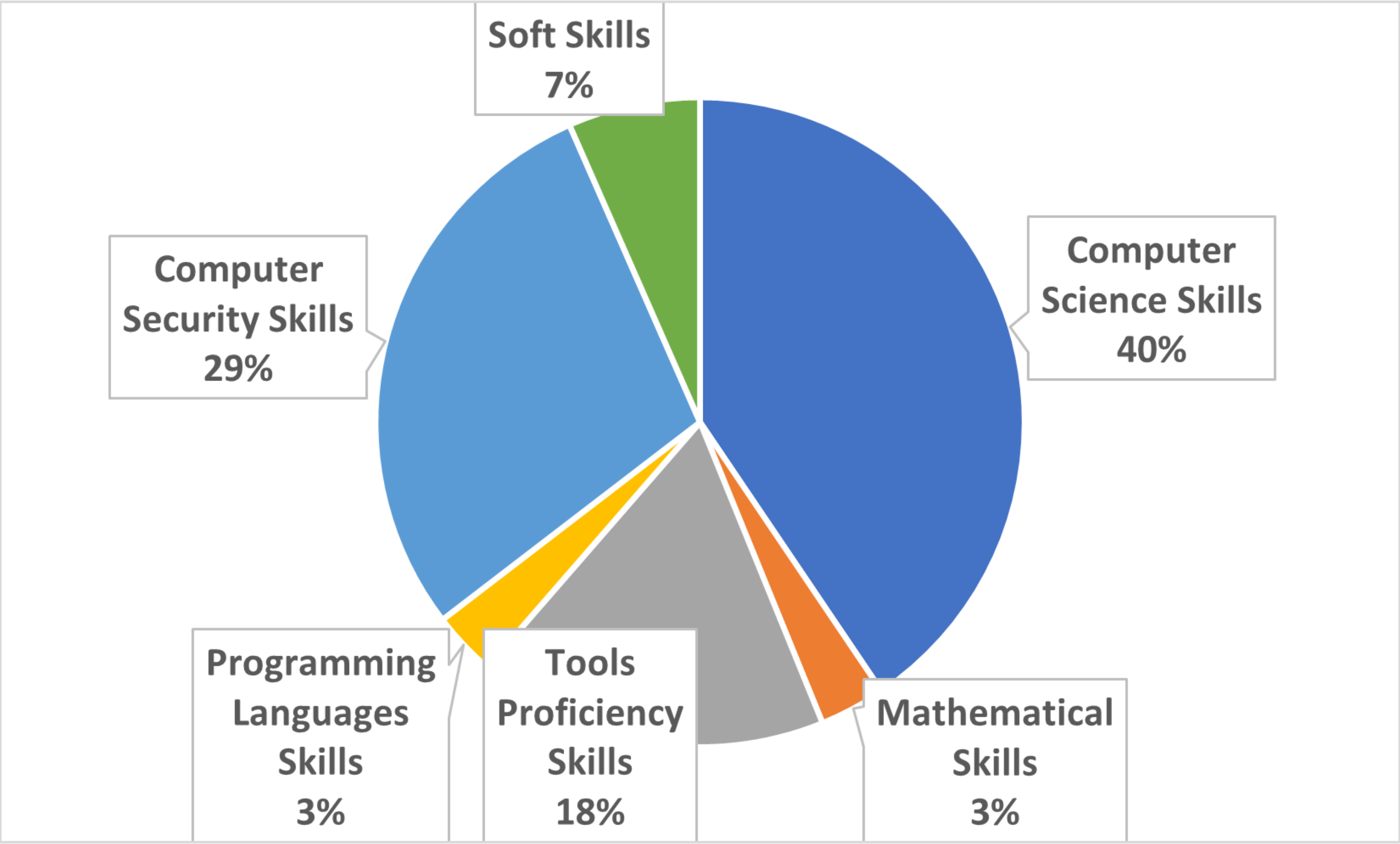


- Segments chosen at beginnings and ends of sentences
- Decreasing token limit to show how more segmentation leads to unpredictable but usually negative results
- Token is an atomic unit of information for an LLM. It is usually either a word or part of a word
- Accuracies: 1500: 87%, 1000: 92%, 500: 83%

<u>Insights:</u>

- While segmentation can improve accuracy, it is important to segment intentionally in order to yield positive results
- Random segmentation generally leads to negative results
- At 1500 tokens, approximately 1.5 segmentations per document. At 1000 approximately 2 segmentations per document, and at 500 approximately 3.5 segmentations per document
- More segmentation leads to more results but usually less accurate

Chat-GPT Automatically Clusters Skills Based on Topic



<u>Insights:</u>

- Most skills are either Computer Science and Computer Security
- Low counts does not correlate to low emphasis(large number of distinct computer science skills, while distinct number of soft skills is lower)
- If a cluster is missing from extraction, it is trivial to add a new category

Conclusion

- Chat-GPT can be used to automate difficult NLP tasks
- With the right prompt, Chat-GPT classification is state of the art
- Prompt Engineering can be used to automate the prompt construction process
- Few-Shot examples require segmentation due to token limits
- If not done properly, segmentation can reduce accuracy

Next Steps

- Train a model to appropriately segment documents
- Use techniques such as few shot examples and prompt engineering to optimize accuracy
- Apply extraction techniques across larger manually labeled datasets
- Create database of sample students to compare compatibilities between different institutions and job postings

Thank You!

Any Questions