

# PDF Mirage: Content Masking Attack Against Information-Based Online Services

Ian Markwood\*, Dakun Shen\*, Yao Liu, and Zhuo Lu  
University of South Florida

\*Co-first authors

Presented by Ian Markwood



# Outline

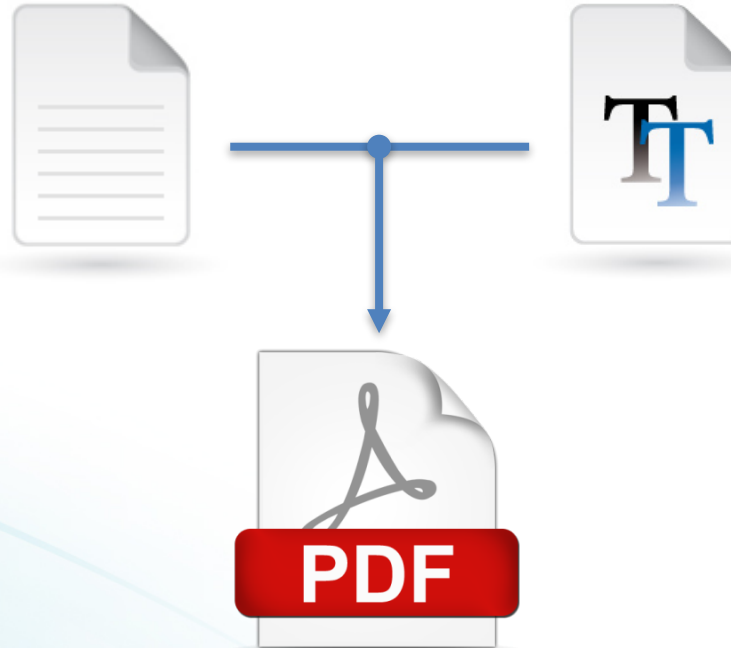
- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion

# Motivation

- The Adobe Portable Document Format (PDF) is the standard for consistent cross-computer document rendering
- PDF documents cannot be edited with commonly accessible tools (MS Word, Adobe Reader, etc.)
- This confers a sense of integrity to the document for the end user

# Motivation

- There is a disconnect between the content of a PDF and what is actually displayed
- A computer and a human see two different things



# Motivation

- Within this disconnect we can perform a content masking attack which compromises the content integrity of PDF files
- Three information-based online systems rely on the integrity of PDF documents:
  - Automatic reviewer assignment systems for academic papers
  - Plagiarism detection systems
  - Search engines

# Outline

- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion

# Background Information

- What do these services have in common?
  - They support PDF submission
  - They scrape the text out of submitted PDF files to perform their function, rather than using Optical Character Recognition (OCR)
  - Text scraping copies the plaintext out of all strings within the PDF file
  - Ignores font associated with text



# Background Information

- Automatic conference reviewer assignment systems
  - Use topic matching to assign reviewers to submitted papers
  - Compare frequent words appearing in reviewers' published papers to frequent words appearing in submitted papers
  - INFOCOM uses Latent Semantic Indexing (LSI)



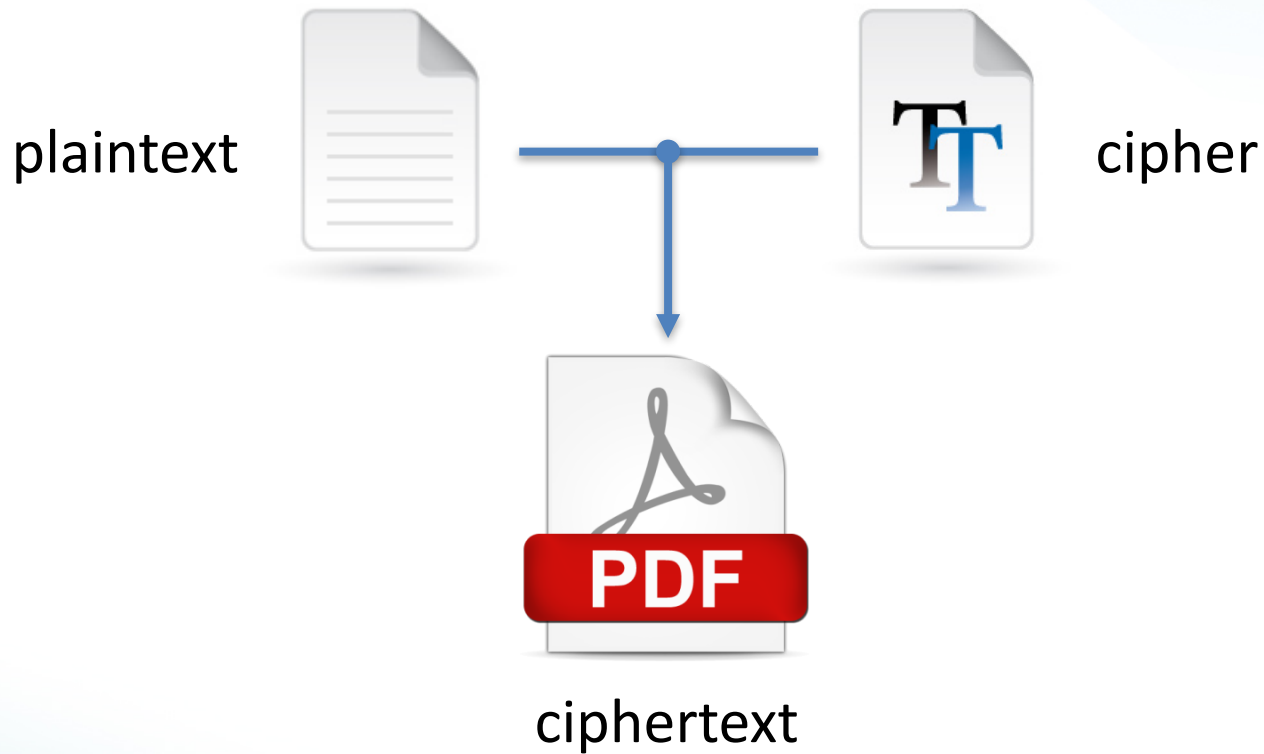
# Background Information

- Plagiarism detection systems
  - Measure similarity between strings within subject document and all other documents submitted thus far
- Document indexing
  - Search engines return documents based on the similarity of their content to the search string

# Outline

- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion

# Content Masking Attack



# Content Masking Attack

- “Masking font” – a custom font with some rearrangement of the character/glyph relationship
- Open source tools such as Font Forge allow copy/paste of character glyphs within fonts
- Custom fonts may be imported into L<sup>A</sup>T<sub>E</sub>X

# Outline

- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion

# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

- An author can target a specific reviewer by replacing enough key words in the paper with key words from the reviewer's papers
- Key words – uncommon words that appear most frequently

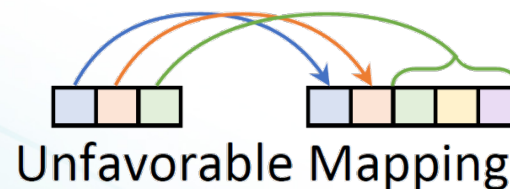
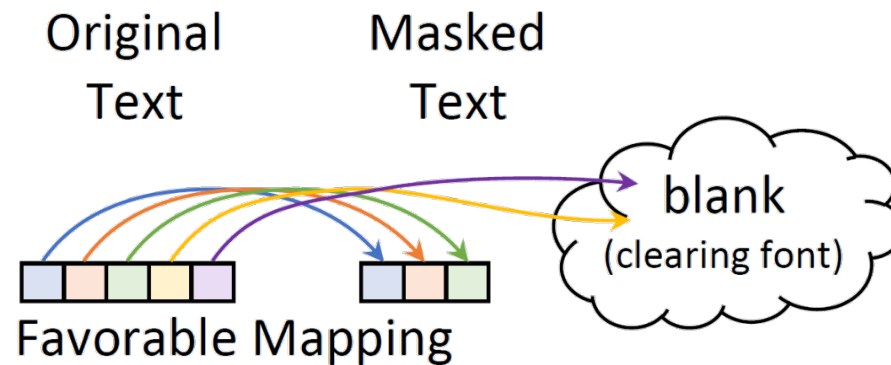
# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

- Algorithm:
  - Order key words in subject paper and target reviewer’s corpus by descending frequency
  - Construct a “word mapping” between these two lists
  - Create a “character mapping” between the letters of each pair of words



# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

- Challenges:
  - One-to-Many Character Mapping
  - Word Length Disparity

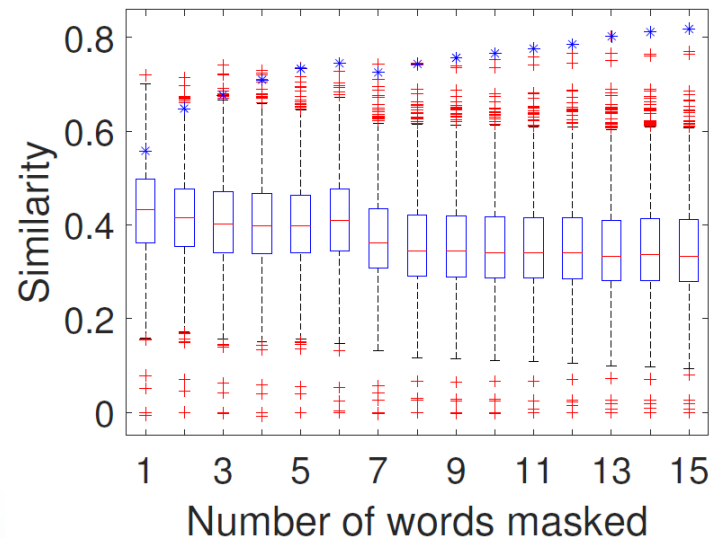


# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

- Experiment:
  - We have reproduced the INFOCOM automatic reviewer assignment system
  - This includes 114 TPC members from a well-known security conference and 2094 of their recently published papers for training
  - 100 additional papers used as testing data

# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

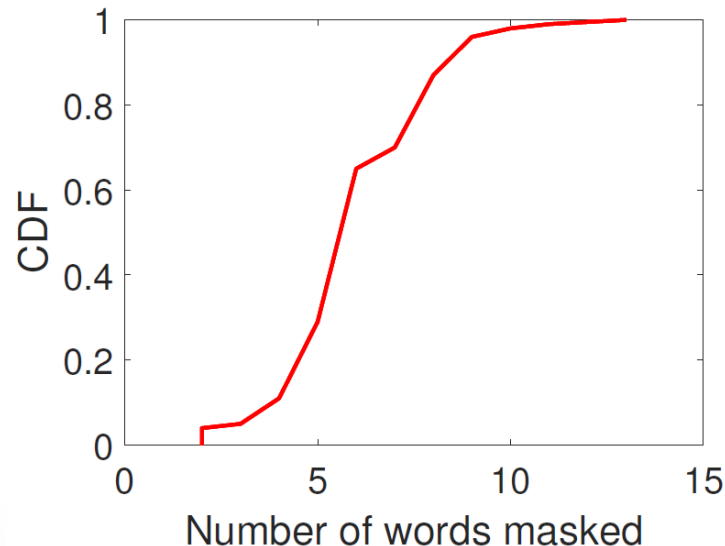
- Experiment:
  - Matching a paper to one reviewer



Similarity scores relative to amount of words masked.  
Blue stars show the desired matching.

# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

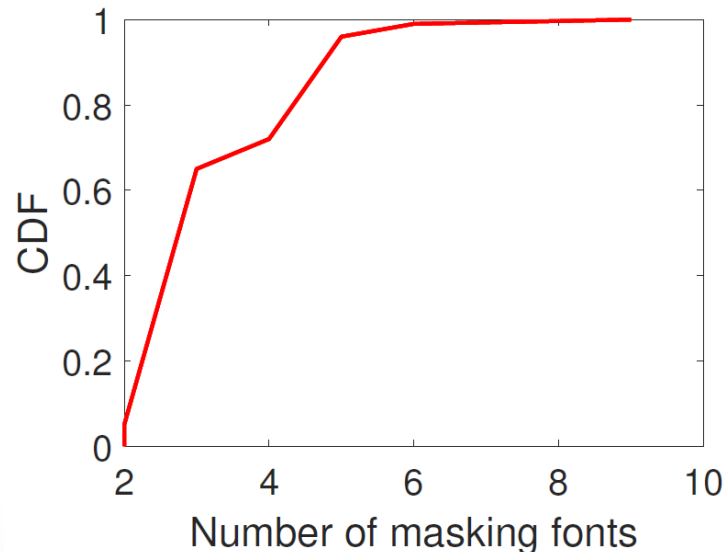
- Experiment:
  - Matching a paper to one reviewer



Word masking requirements for all 100 testing papers

# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

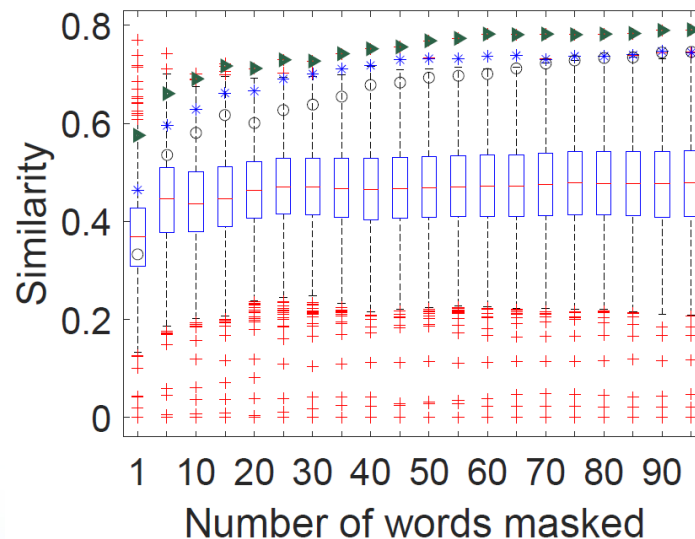
- Experiment:
  - Matching a paper to one reviewer



Masking font requirements for all 100 testing papers

# Content Masking Attack Against Automatic Conference Reviewer Assignment Systems

- Experiment:
  - Matching a paper to multiple reviewers



Similarity scores relative to amount of words masked, between a paper and three reviewers. Blue stars, black circles, and green triangles show the desired matchings

# Outline

- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion



# Content Masking Attack Against Plagiarism Detection

- A cheating student can evade a plagiarism detector by replacing the underlying text with gibberish
- Use a “scrambling font” to render the gibberish as legible (plagiarized) text
- Results in zero similarity with existing work

# Content Masking Attack Against Plagiarism Detection

- Zero similarity is unrealistic due to common phrases in language
- We evaluate three methods to target a specific similarity score
- Each method chooses what text to scramble and what text to leave unaltered

# Content Masking Attack Against Plagiarism Detection

- By letter
  - Use scrambling font which scrambles all characters
  - Remove characters from being scrambled by order of their frequency of appearance in the language
  - Continue removing characters until a target similarity score is reached

# Content Masking Attack Against Plagiarism Detection

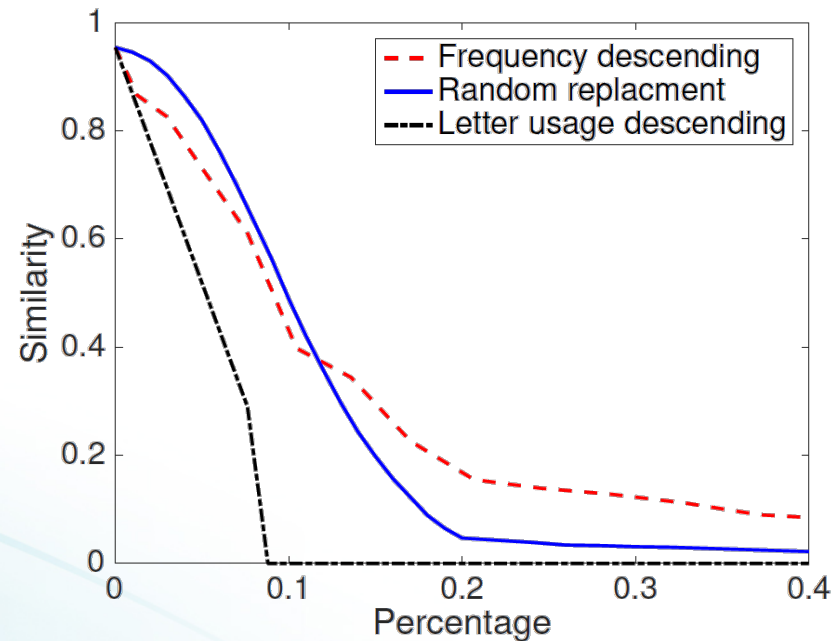
- By word, in frequency of appearance
  - Use scrambling font which scrambles all characters
  - Order distinct words by frequency of appearance
  - Apply scrambling font to all words
  - Remove scrambling font from distinct words until a target similarity score is reached

# Content Masking Attack Against Plagiarism Detection

- By word, at random
  - Use scrambling font which scrambles all characters
  - Iterate over document, applying scrambling font at random according to chosen probability
  - Modify probability until a target similarity score is reached

# Content Masking Attack Against Plagiarism Detection

- Experiment:
  - Apply scrambling fonts to 10 published papers and target 5-15% similarity score measured by Turnitin



# Outline

- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion



# Content Masking Attack Against Document Indexing

- An attacker can place spam or illicit content in PDF documents indexed by search engines
- These PDFs can show ads instead of legitimate content that users search for

# Content Masking Attack Against Document Indexing

- This can be considered a special case of the reviewer assignment system subversion method
- Instead of masking particular words, we are masking the entire document
- Not constrained by spaces however

# Content Masking Attack Against Document Indexing

- The larger number of masked characters requires more masking fonts
- Instead of generating fonts ad hoc, we make one font for each glyph
- ~84 fonts
- Allows for easy automated generation of masked documents

# Content Masking Attack Against Document Indexing

- Experiment
  - Used 5 well-known published papers
  - Masked each as gibberish

Ipf Owhqwvg wn Nwvgqghfvdm bve Azfeqdbhf  
Awdsg qv b Qbhbcbgf Bmghfu

1N.A. Ogkbzby, V.O. Wzbm, S.T. Awzqf, bve L.A. Izbqofz  
LEH Sfgfbzdp Abcwzbhwzm Bbv Vwgf, Nbtqnwzvqb

## Abstract

Lv ebhbcbgf gmghfug, igfzg bddfgg gpbzfc ebhb ivefz hpf bggiuxhqvw hpbh  
hpf ebhb gbhqgnqfg dfzhhqv dwvgqghfvdm dwvghzbqvhg. Ipqg xbxiz efnqvfg  
hpf dwvdfxhg wn hzbvghdqvw, dwvgqghfvdm bve gdpfeitf bve gpwkg hpbh  
dwvgqghfvdm zfyiqzfg hpbh b hzbvghdqvw dbvvh zfyifgh vfk twdsg bnhfz

# Content Masking Attack Against Document Indexing

- Experiment
  - Submitted them to leading search engines for indexing (Google, Bing, Yahoo!, DuckDuckGo)
  - Results were the same for all test documents

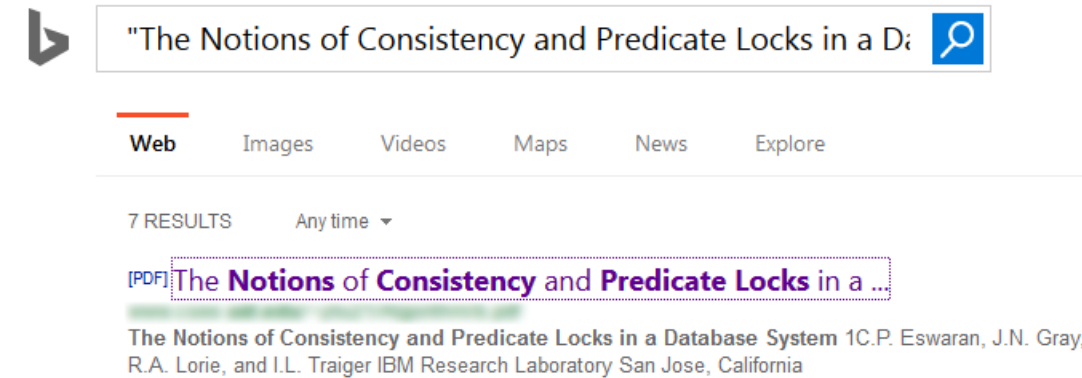
# Content Masking Attack Against Document Indexing

- Experiment

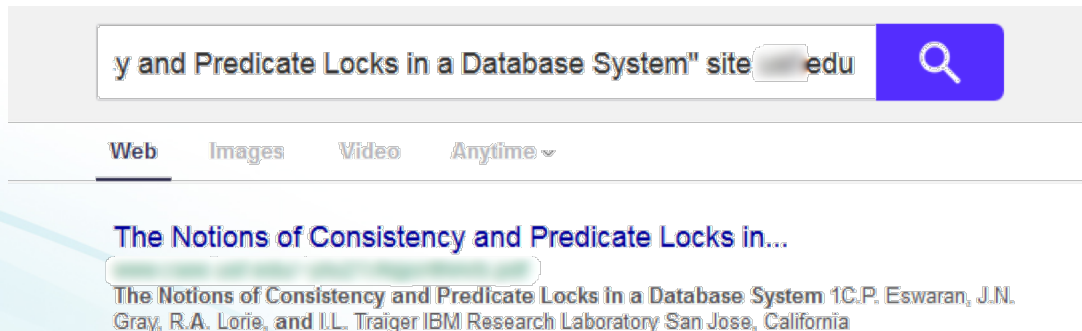
Search Engine	Indexed Papers	Attack Successful	Evades Spam Detection	Not Later Removed
Google	✓	✗	✗	✗
Bing	✓	✓	✓	✓
Yahoo!	✓	✓	✗ → ✓	✓
DuckDuckGo	✓	✓	✓	✓

# Content Masking Attack Against Document Indexing

- Experiment



A screenshot of a search engine interface. The search bar contains the text "The Notions of Consistency and Predicate Locks in a Database System" followed by a magnifying glass icon. Below the search bar, there are tabs for "Web", "Images", "Videos", "Maps", "News", and "Explore". The "Web" tab is selected. Below the tabs, it says "7 RESULTS" and "Any time". The first result is a PDF document titled "The Notions of Consistency and Predicate Locks in a Database System" by 1C.P. Eswaran, J.N. Gray, R.A. Lorie, and I.L. Traiger. The authors' names and affiliation are listed below the title.



A screenshot of a search engine interface showing a content masking attack. The search bar contains the text "y and Predicate Locks in a Database System" site [redacted] .edu followed by a magnifying glass icon. Below the search bar, there are tabs for "Web", "Images", "Video", and "Anytime". The "Web" tab is selected. The first result is a document titled "The Notions of Consistency and Predicate Locks in a Database System" by 1C.P. Eswaran, J.N. Gray, R.A. Lorie, and I.L. Traiger. The authors' names and affiliation are listed below the title.



# Outline

- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion

# Content Masking Defense

- One feasible defense: perform Optical Character Recognition (OCR) on the document to check the integrity of each character.
- Problem:
  - High computational overhead
  - High false positive rate



50,000 - 75,000  
characters

# Content Masking Defense – Our proposal

- Render each character in the fonts embedded in the subject PDF file and perform OCR on those character codes rather than the rendered PDF file itself.
- Save processing time



50,000 - 75,000  
characters



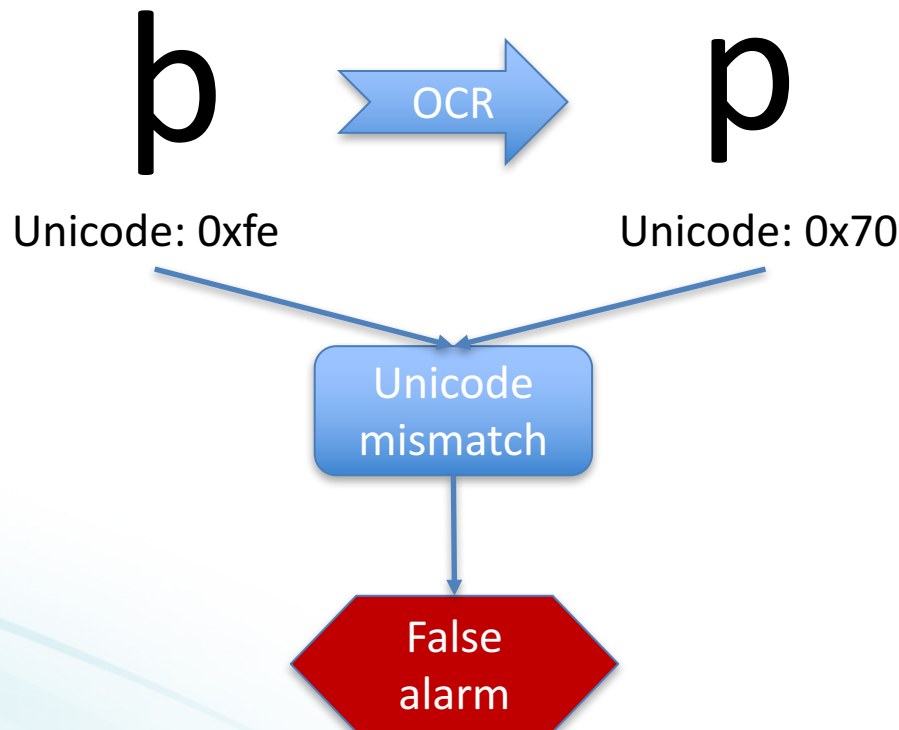
100 -2000  
characters

# Challenges and Technical Details

- **Challenge 1:** Whole font file is embedded
  - Contain  $2^{16} = 65,536$  characters maximum
  - Cause high computational overhead
- **Solution:** Scan the document to extract the characters used, and perform OCR on the series of character used in each font.

# Challenges and Technical Details

- **Challenge 2: Special characters**



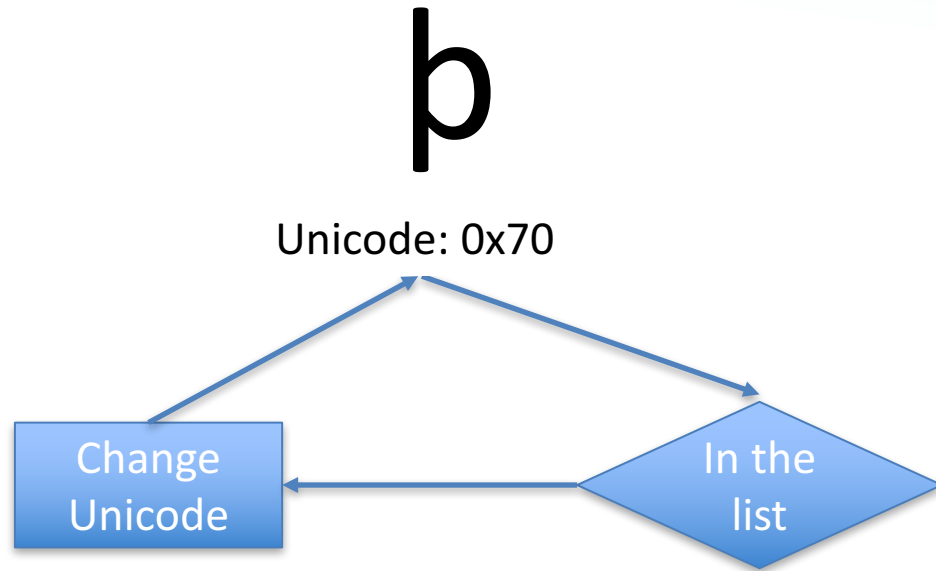
# Challenges and Technical Details

- **Solution: Font Training**

1. Perform OCR on the font and list all similar characters.
2. If the detected glyph is in the similar character list, replace the character's Unicode as the normal letter it looks like.

# Font Training

White list	
ã 0xe3	a 0x61
ħ 0x267	h 0x68
Ƶ 0x460	W 0x57
.....	.....
þ 0xfe	p 0x70
.....	.....

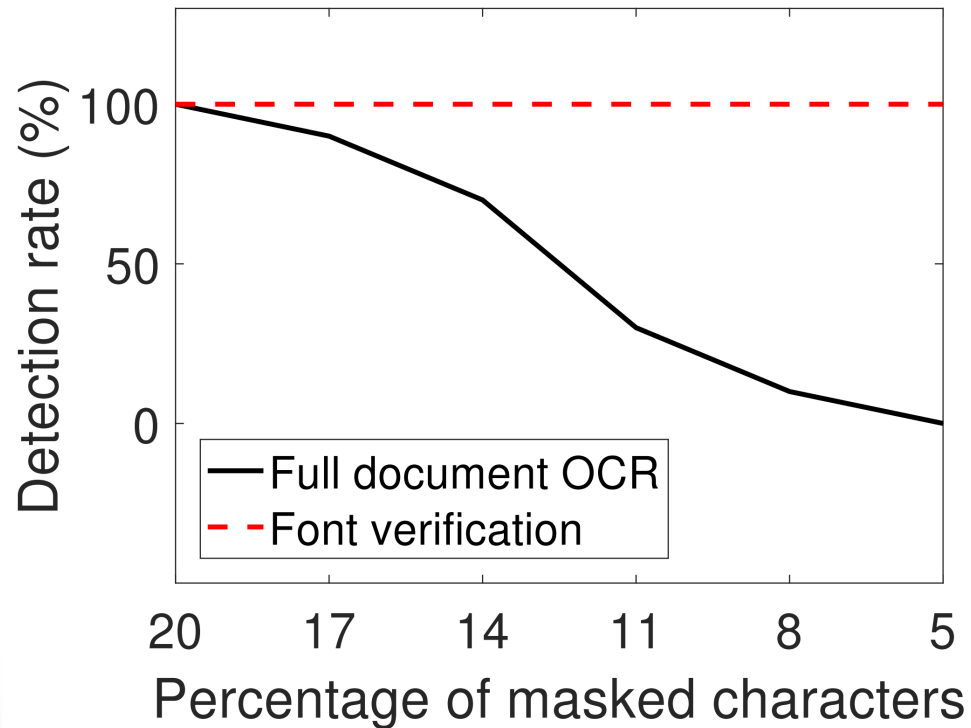




# Font Verification Performance

- Experiment 1
  - To analyze the **accuracy** of our Font Verification method and the Whole Document OCR method
  - Generated 10 PDF files with masked characters varying from 5-20% in frequency of appearance

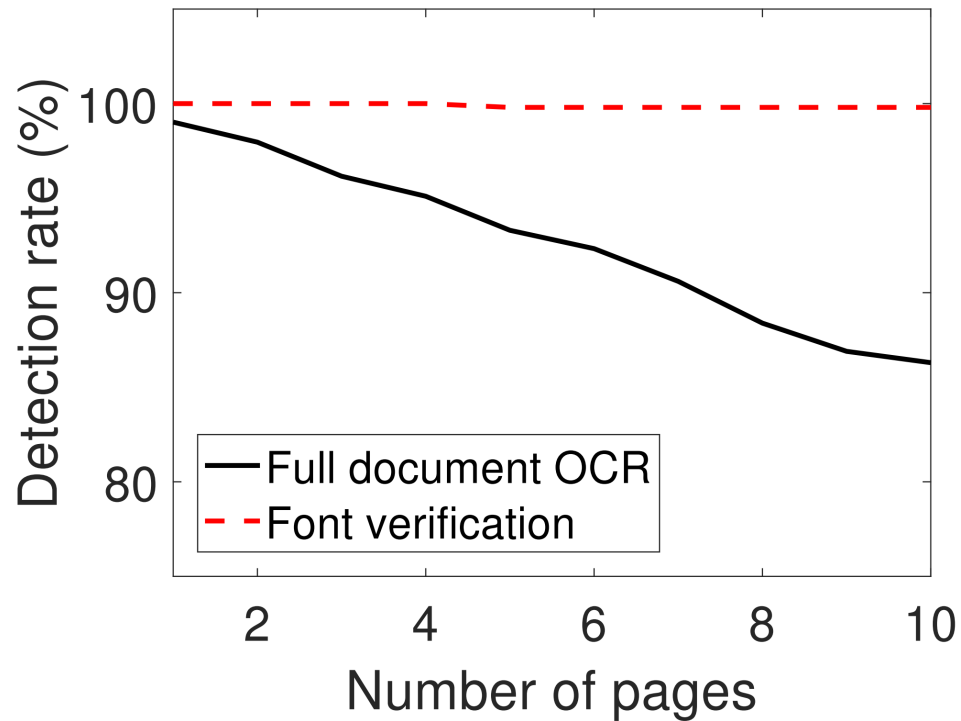
# Performance – Experiment 1



# Font Verification Performance

- Experiment 2
  - To analyze the effects of **document length** on the **detection rate** for each method.
  - Generated 10 PDF files ranging from 1-10 pages in length and having an even 30% distribution of masked characters

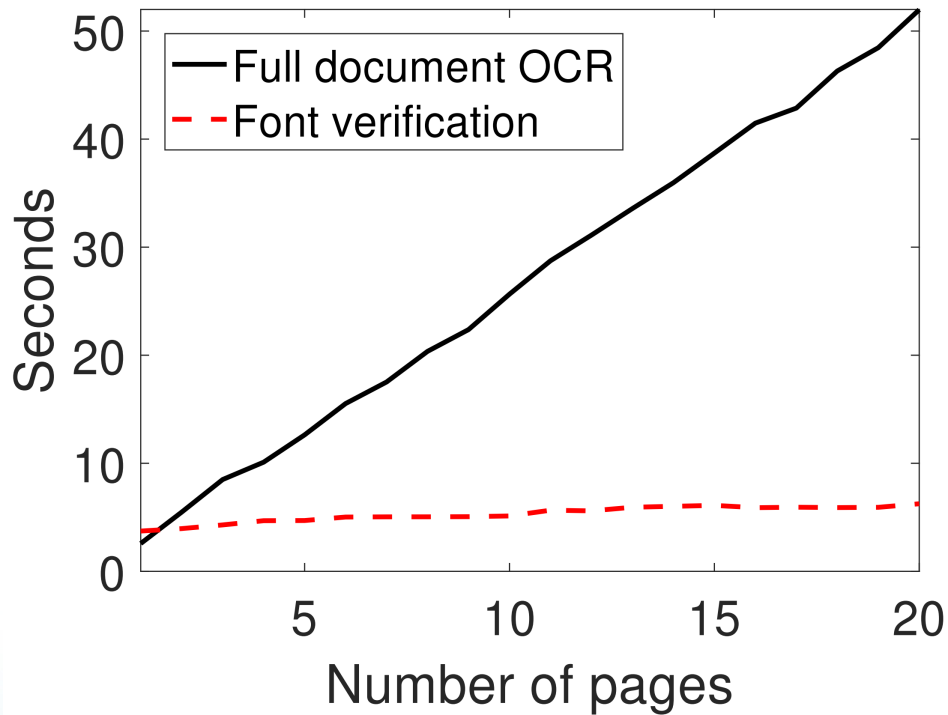
# Performance – Experiment 2



# Font Verification Performance

- Experiment 3
  - To analyze the effect of **document length** on the **detection time** for each method
  - Generated 20 PDF files ranging from 1-20 pages in length and having a 30% distribution of masked characters

# Performance – Experiment 3



# Outline

- Motivation
- Background Information
- Content Masking Attack
  - Against Conference Reviewer Assignment Systems
  - Against Plagiarism Detection
  - Against Document Indexing
- Content Masking Defense
- Conclusion



# Conclusion

- We describe a new content masking attack against the Adobe PDF standard
- We create and evaluate algorithms for effectively performing attacks against:
  - Automatic reviewer assignment systems
  - Plagiarism detection
  - Document indexing
- We create and evaluate a font verification algorithm that is more accurate and lightweight than OCR

# Thank you!

- Questions?

PDF file image from <http://iconbug.com/detail/icon/5940/file-format-pdf/>

True Type font file image from <https://typography.guru/journal/opentype-myths-explained-r24/>

