Where are the Facts? Searching for Fact-checked information to Alleviate the Spread of Fake News

Nguyen Vo, Kyumin Lee (2020)

ECE 594 (Spring 2022)

Paper Presentation

Rakesh Vaideeswaran

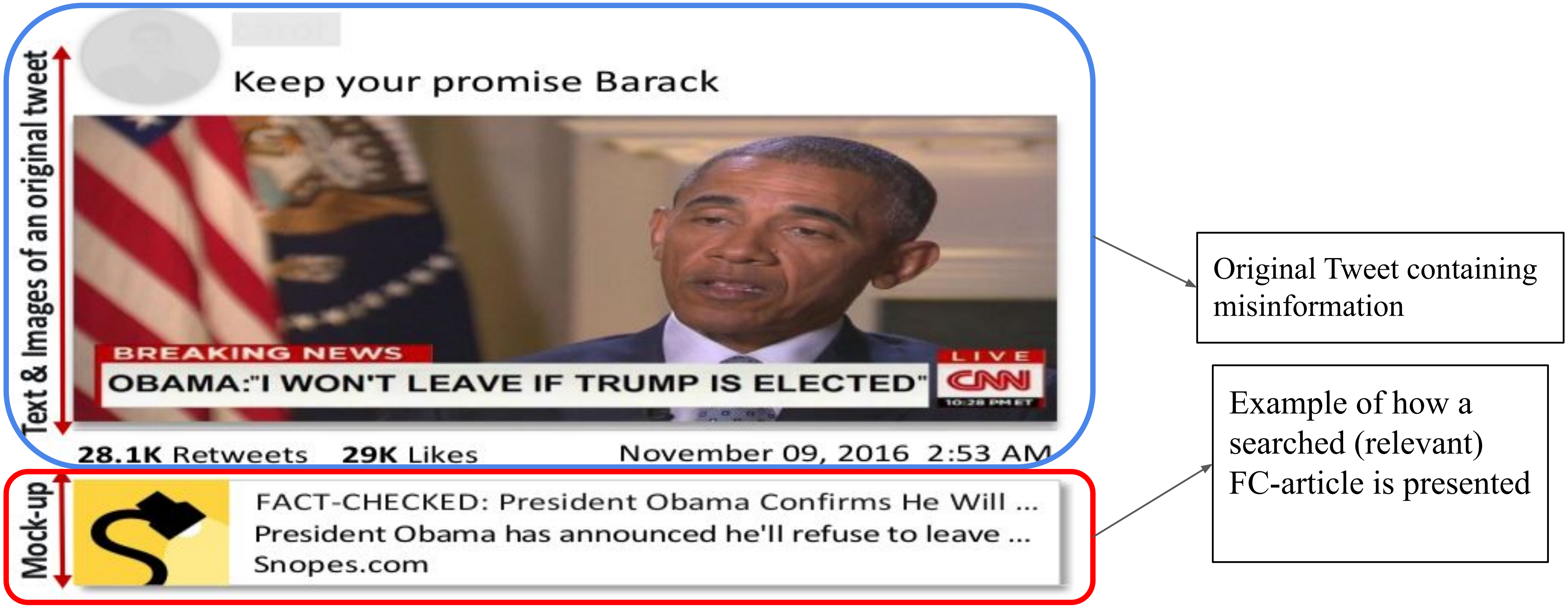
04/14/2022

Background

- Fake news is prevalent in social media
- Fact-checking systems exist:
    - Focus only on fact-checking
    - Neglect online users who spread misinformation
- Since 2014, number of fact-checking systems has **increased by 400% in 60 countries**[1]
- Fake news is still prevalent
- Cause citizens' misperception about political candidates, threatened public health, etc., which is very concerning

[1]Mark Stencel. 2019. Number of fact-checking outlets surges to 188 in more than 60 countries. https://bit.ly/36y3S3l

Example



By Incorporating **Fact Checking (FC)-article** with social media posts:

1. Users can be warned about fake news
2. Increased volume of verified content

Contributions

- Searching Fact-Checking (FC) articles to increase user awareness of fact-checked information
- Novel Neural Ranking model that uses both textual and visual information (integrated attention mechanism)
- Perform experiments on two datasets, and demonstrate effectiveness and generality over existing document ranking methods

Challenges

What information in original tweets should be used to find correct FC-articles?

- Using **only text** from original tweets is **suboptimal**
- Authors propose to use information from **both text** and **images**

How can a framework be designed that retrieves and ranks FC-articles?

- *Step 1*: Basic retrieval (BM25) to find initial lists of candidate FC-articles (using information from original tweet: a) ***text (BM25-T),*** b) ***image (BM25-I),*** c) ***text in image (BM25-TI)***)
- *Step 2*: Re-rank the lists obtained in Step 1 (attention mechanism - to integrate textual and visual information)

Framework: Inputs

Original Tweet $\boldsymbol{q} : (\boldsymbol{q}_{text}, \boldsymbol{q}_{images})$

$\boldsymbol{q}_{text}$ - sequence of N words $\{w_1, w_2, \ldots\ldots., w_N\}$

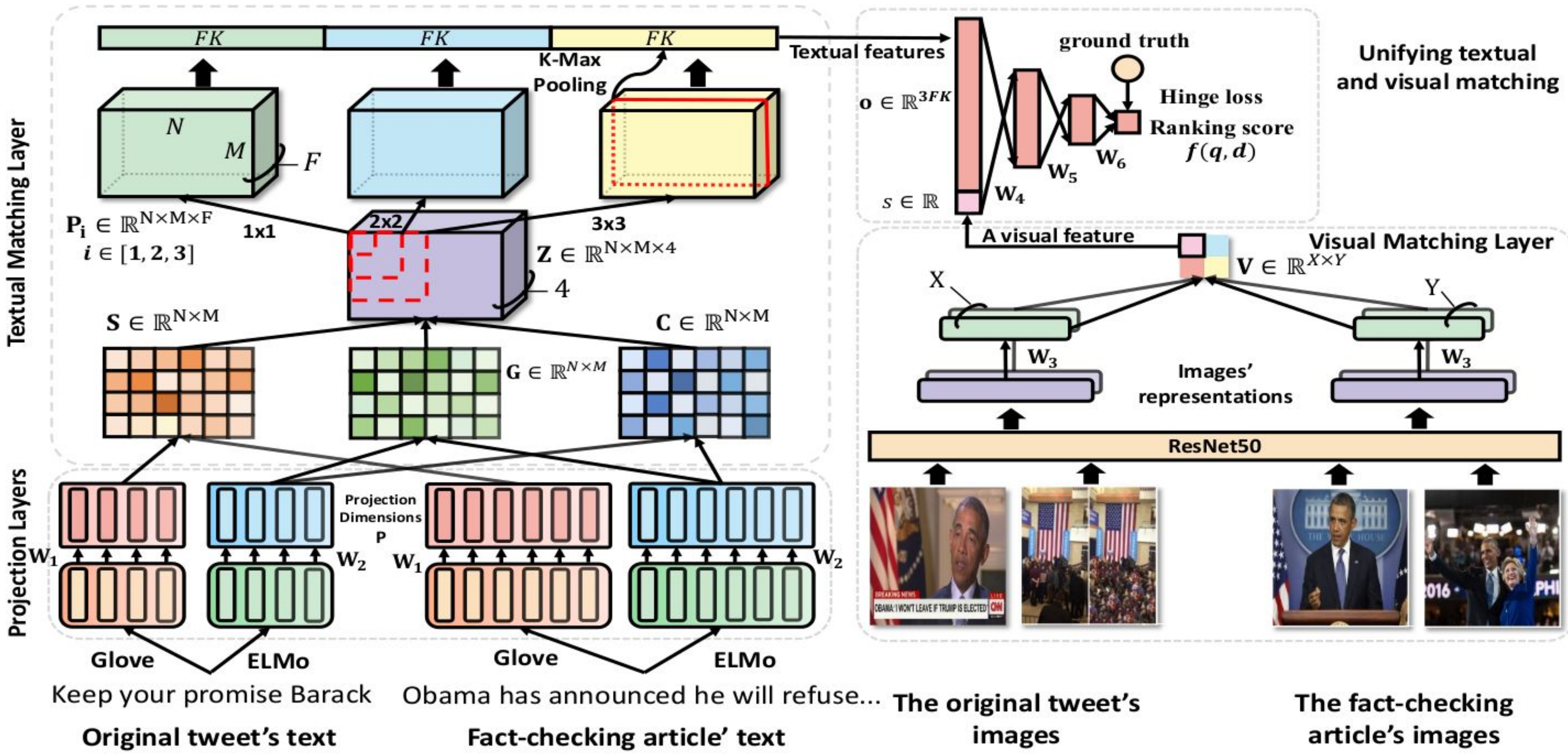
$\boldsymbol{q}_{images}$ - list of X images $\{v_1, v_2, \ldots\ldots., v_X\}$

FC-article $\boldsymbol{d} : (\boldsymbol{d}_{text}, \boldsymbol{d}_{images})$

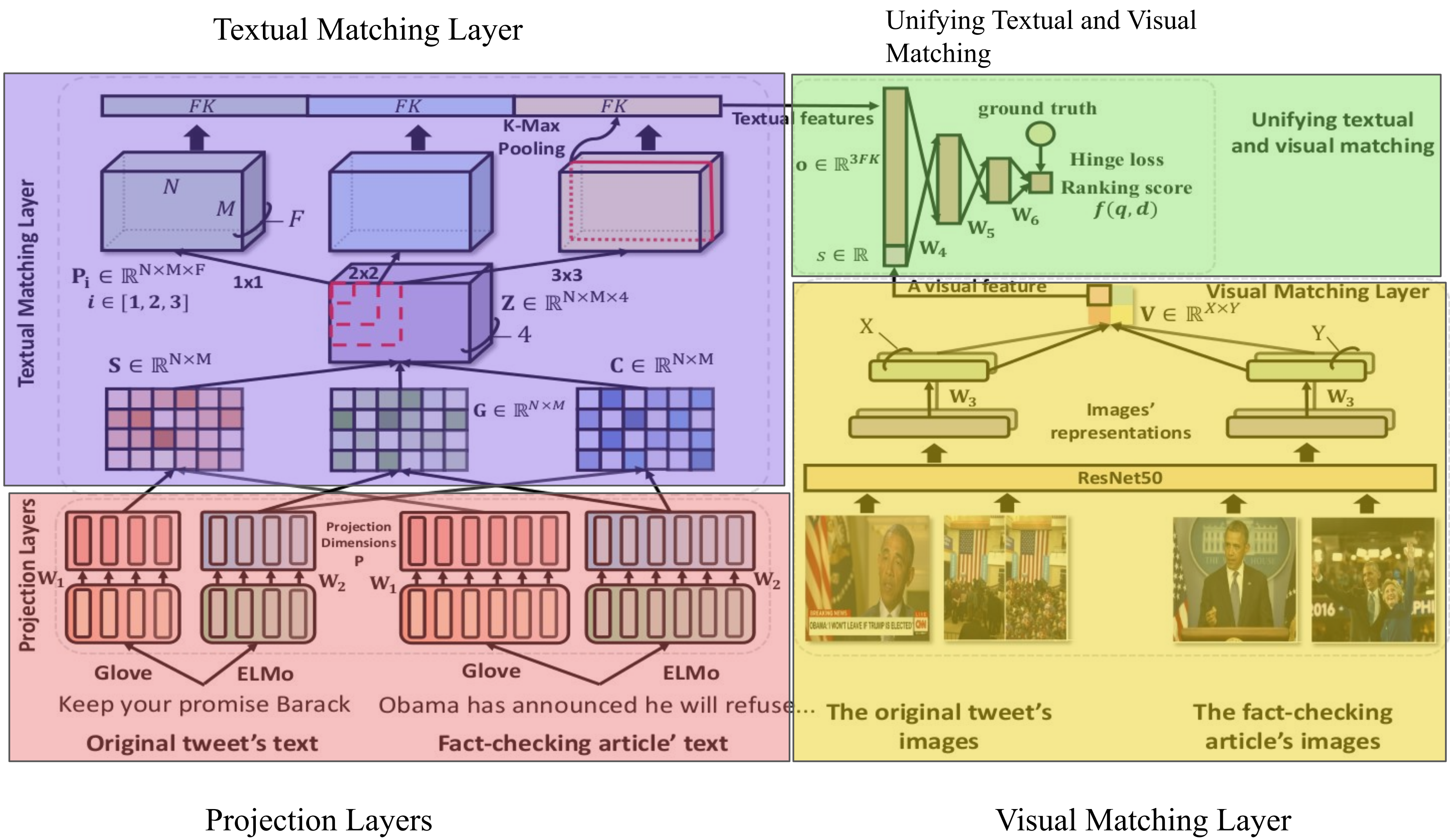
$\boldsymbol{d}_{text}$ - sequence of M words $\{w_1, w_2, \ldots\ldots., w_M\}$

$\boldsymbol{d}_{images}$ - list of Y images $\{v_1, v_2, \ldots\ldots., v_Y\}$

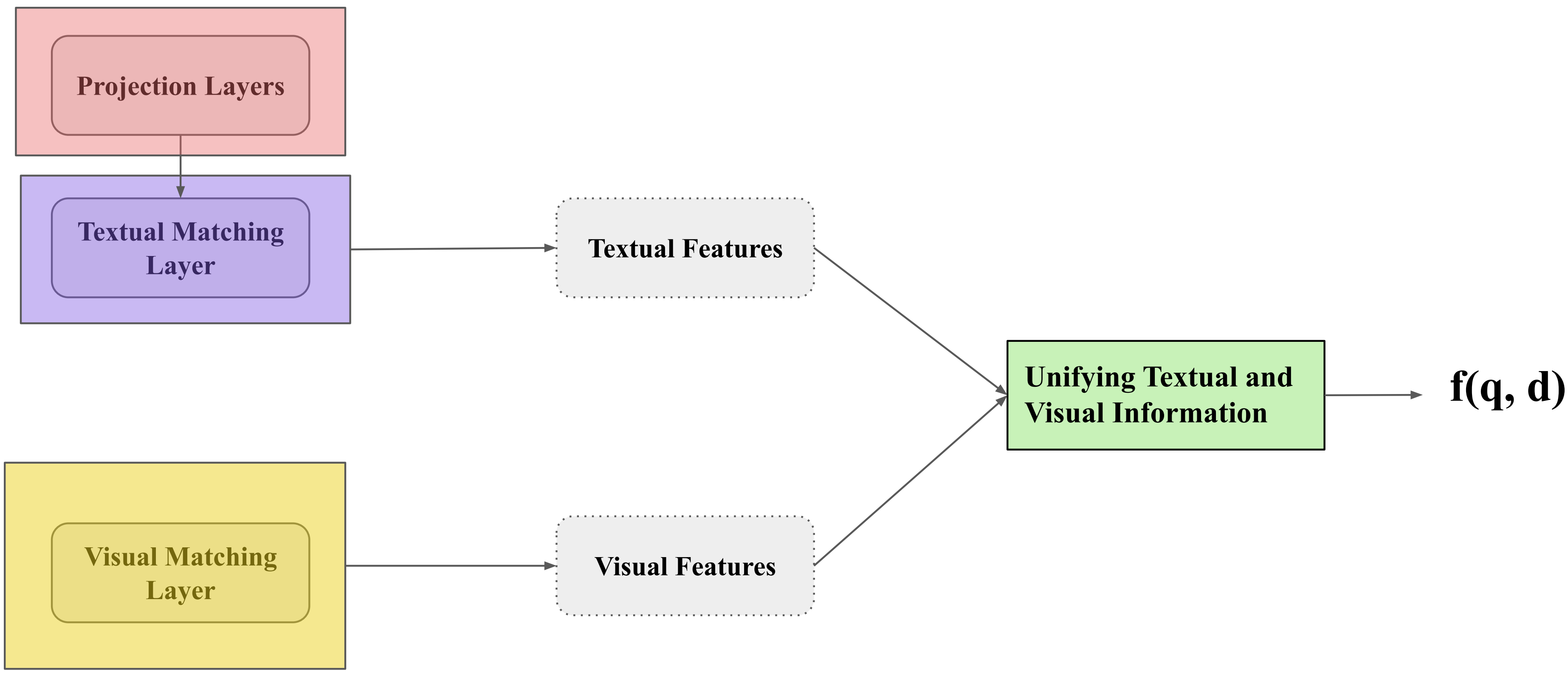
Aim is to derive a mapping ***f(q, d)*** [***ranking function*** - used to rank FC-articles]



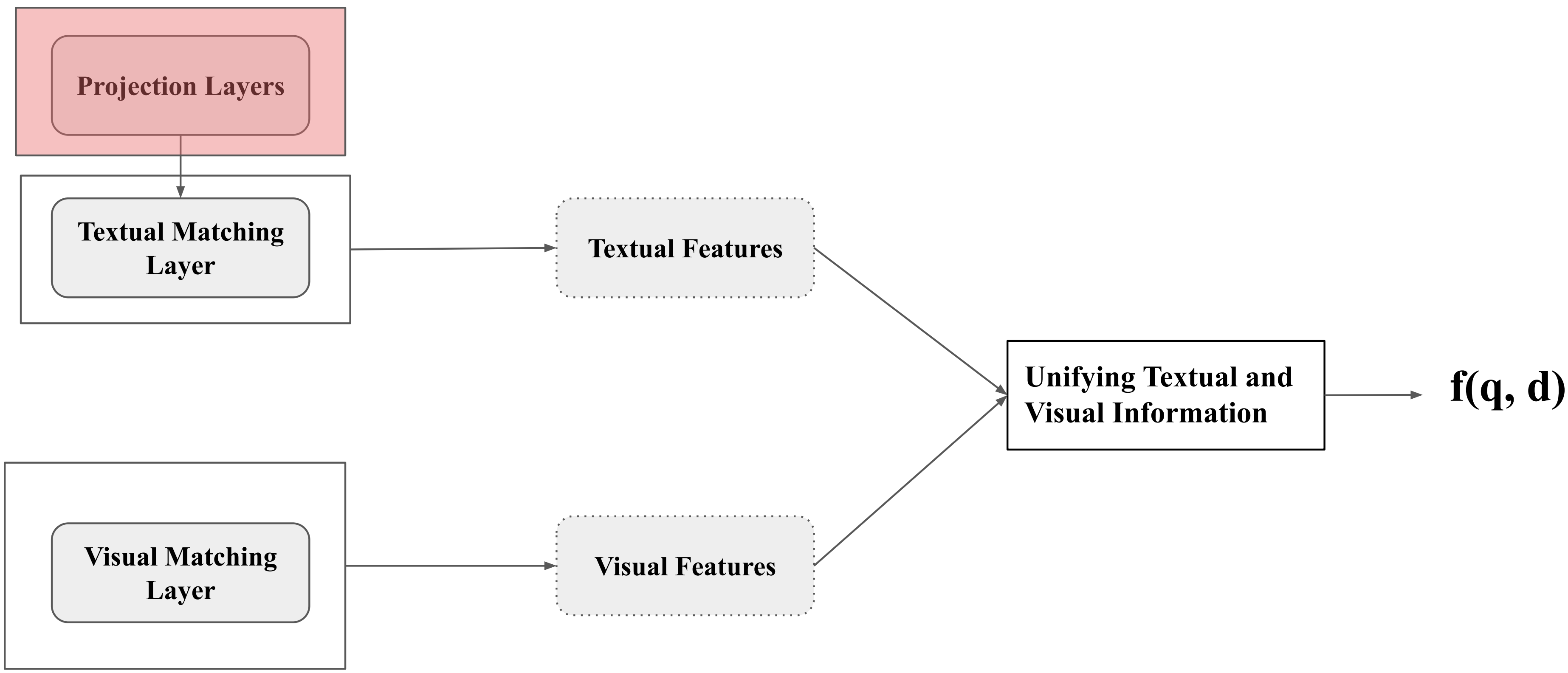
Textual Matching Layer
Projection Layers
FK
FK
FK
K-Max
Pooling
N
M
F
$\mathbf{P_i} \in \mathbb{R}^{N\times M\times F}$
$i \in [1,2,3]$
1x1
2x2
3x3
$\mathbf{Z} \in \mathbb{R}^{N\times M\times 4}$
4
$\mathbf{S} \in \mathbb{R}^{N\times M}$
$\mathbf{G} \in \mathbb{R}^{N\times M}$
$\mathbf{C} \in \mathbb{R}^{N\times M}$
Projection
Dimensions
P
W_1
W_2
W_1
W_2
Glove
ELMo
Glove
ELMo
Keep your promise Barack
Obama has announced he will refuse...
Original tweet's text
Fact-checking article' text
Textual features
$\mathbf{o} \in \mathbb{R}^{3FK}$
$s \in \mathbb{R}$
W_4
W_5
W_6
ground truth
Hinge loss
Ranking score
$f(q,d)$
Unifying textual
and visual matching
A visual feature
Visual Matching Layer
$\mathbf{V} \in \mathbb{R}^{X\times Y}$
X
Y
W_3
W_3
Images'
representations
ResNet50
BREAKING NEWS
OBAMA: I WON'T LEAVE IF TRUMP IS ELECTED
CNN
2016
The original tweet's
images
The fact-checking
article's images


Textual Matching Layer
Unifying Textual and Visual Matching
Textual Matching Layer
FK
FK
FK
K-Max Pooling
Textual features
N
M
F
$P_i \in \mathbb{R}^{N\times M\times F}$
$i \in [1,2,3]$
1x1
2x2
3x3
$Z \in \mathbb{R}^{N\times M\times 4}$
4
$S \in \mathbb{R}^{N\times M}$
$G \in \mathbb{R}^{N\times M}$
$C \in \mathbb{R}^{N\times M}$
Projection Layers
Projection Dimensions P
W_1
W_2
W_1
W_2
Glove
ELMo
Glove
ELMo
Keep your promise Barack
Obama has announced he will refuse…
Original tweet's text
Fact-checking article' text
ground truth
Unifying textual and visual matching
$o \in \mathbb{R}^{3FK}$
Hinge loss
Ranking score $f(q,d)$
W_4
W_5
W_6
$s \in \mathbb{R}$
A visual feature
Visual Matching Layer
$V \in \mathbb{R}^{X\times Y}$
X
Y
W_3
W_3
Images' representations
ResNet50
The original tweet's images
The fact-checking article's images
Projection Layers
Visual Matching Layer

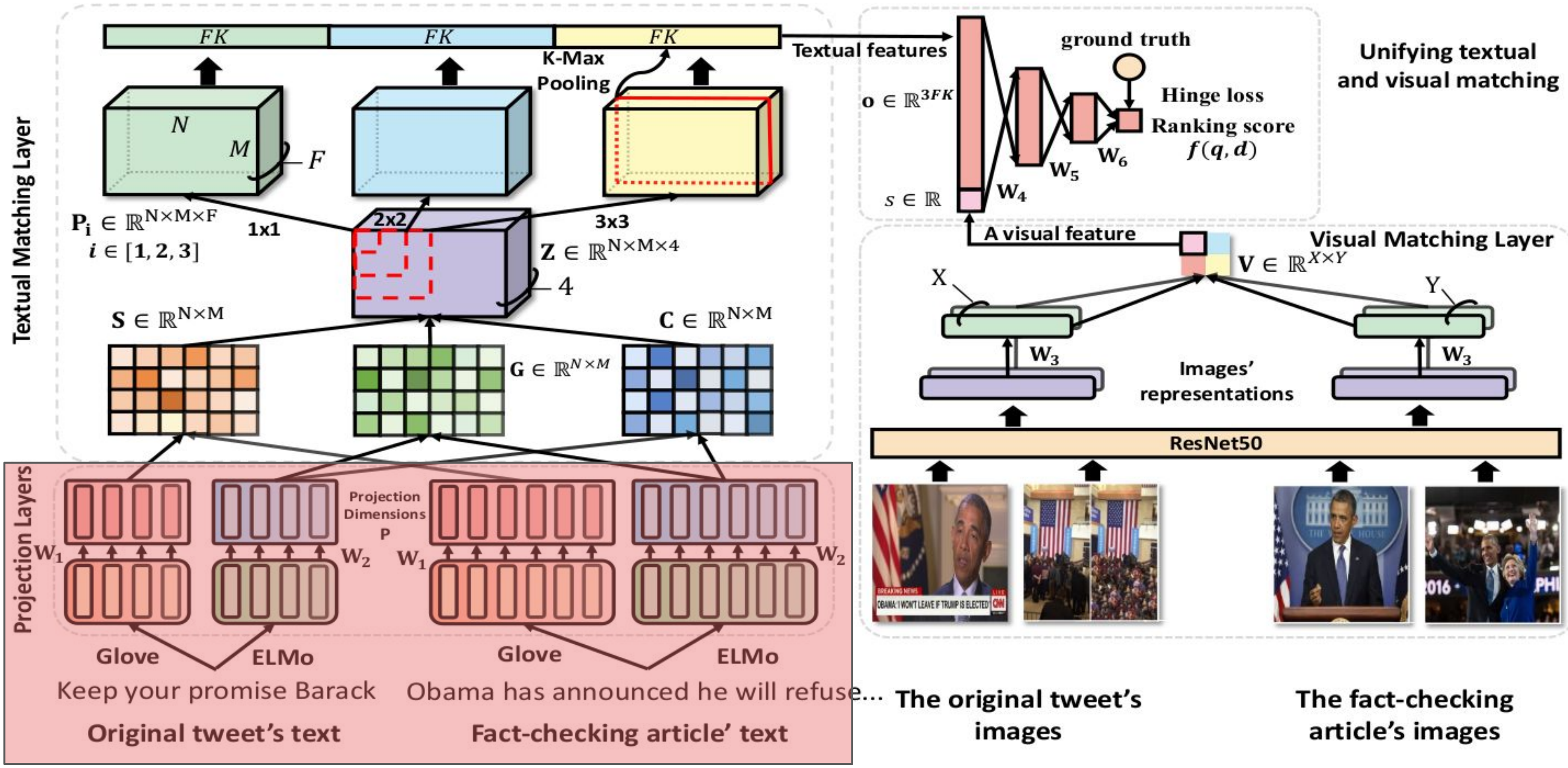
Framework: Multimodal Attention Network - MAN



Framework: Multimodal Attention Network - MAN

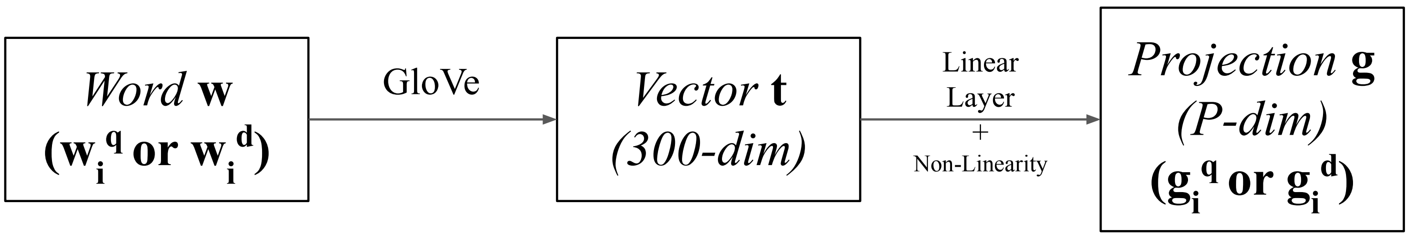


Projection Layers



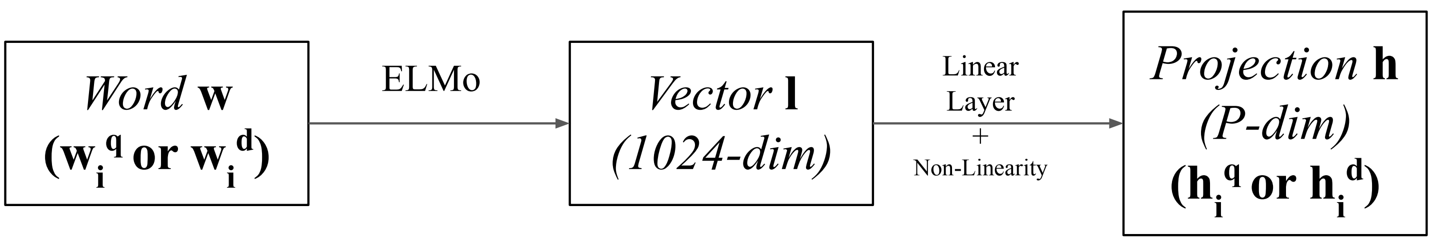
Projection Layers

1. Projection Layer for GloVe Embeddings



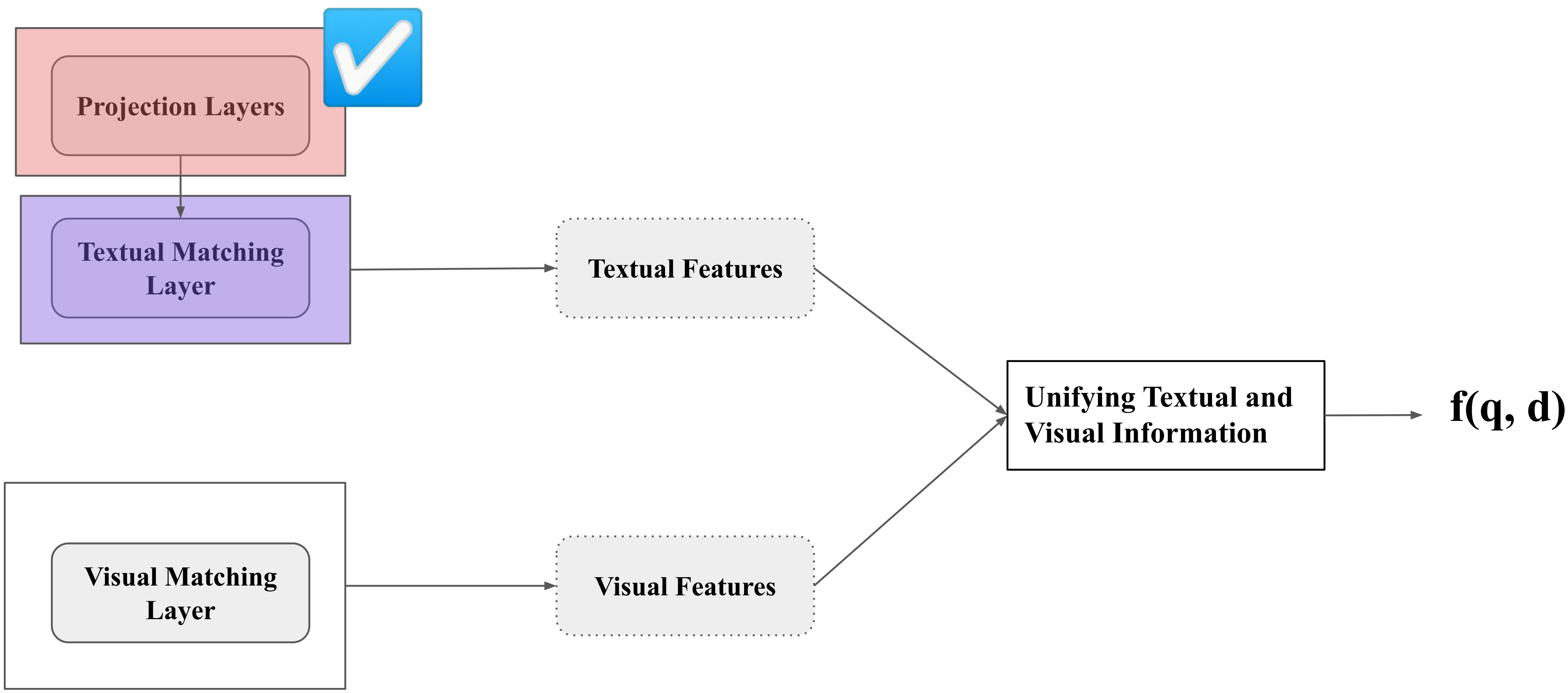
P = 256

2. Projection Layer for Contextual Word Embeddings

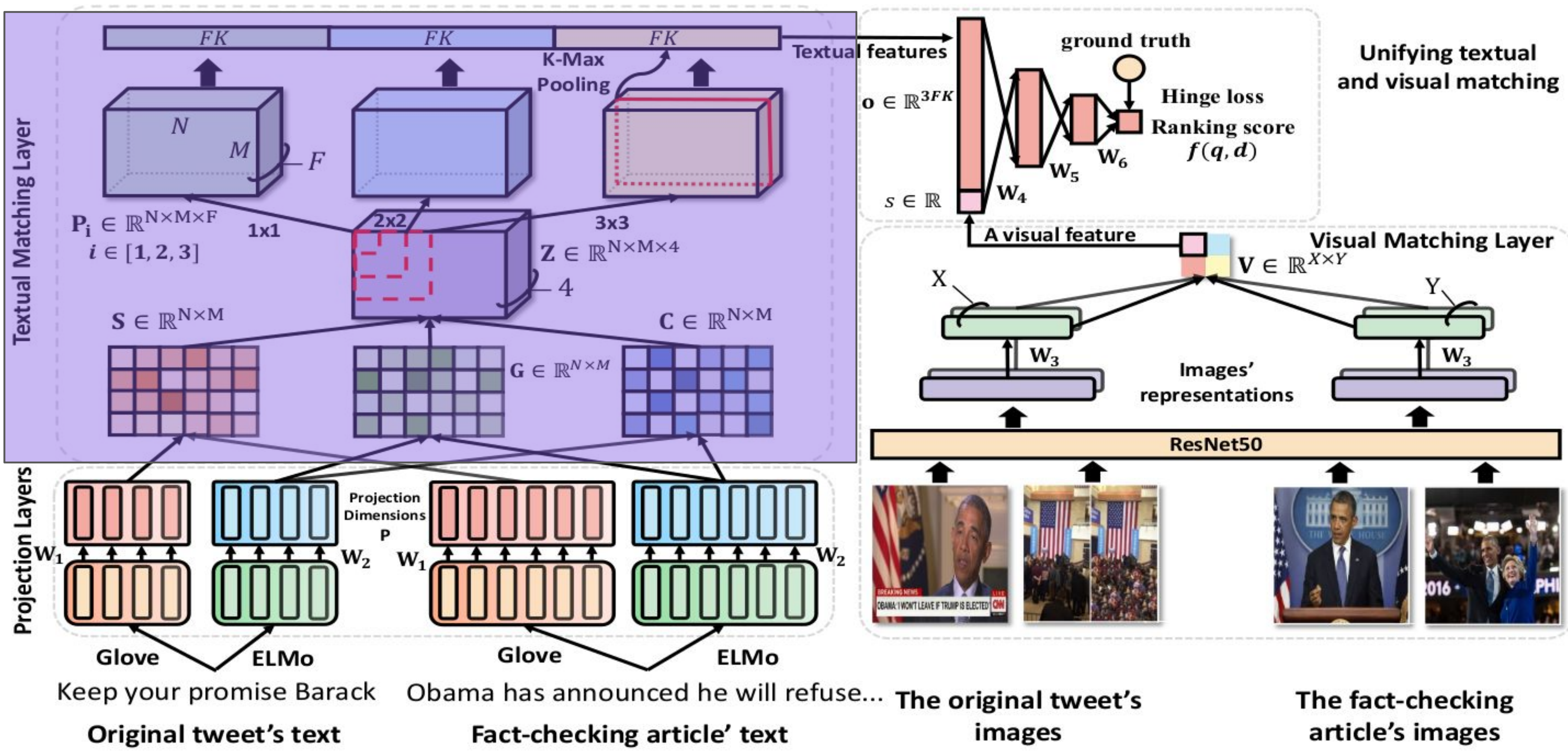


P = 256

Framework: Multimodal Attention Network - MAN



Textual Matching Layer



Textual Matching Layer

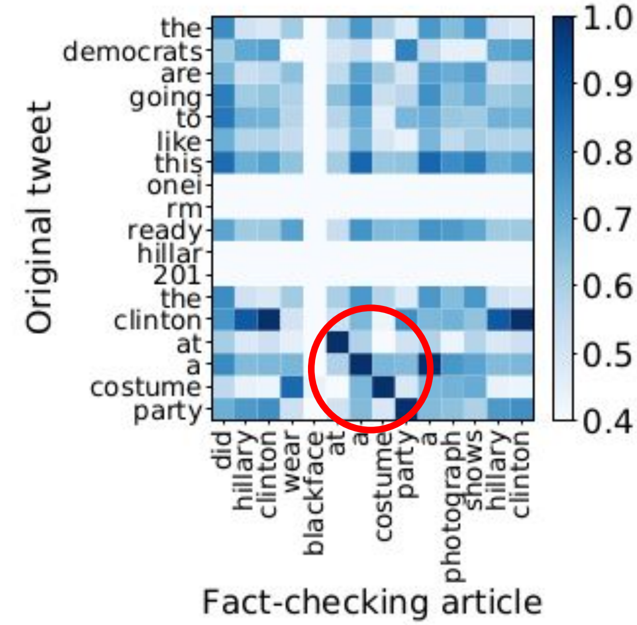
1. GloVe Embedding Interactions
2. ELMo Embedding Interactions
3. Attended Interaction Matrix

GloVe Embedding Interactions

An article is relevant to the original tweet if they have overlapping or similar words

$$\mathbf{S}_{ij} = \frac{\mathbf{g}_i^{q^T} \cdot \mathbf{g}_j^d}{||\mathbf{g}_i^q|| \times ||\mathbf{g}_j^d||}, \mathrm{i} = 1..\mathrm{N}, \mathrm{j} = 1..\mathrm{M}$$

Recall that $\mathbf{g}$ is the output of the Projection Layer for GloVe Embedding



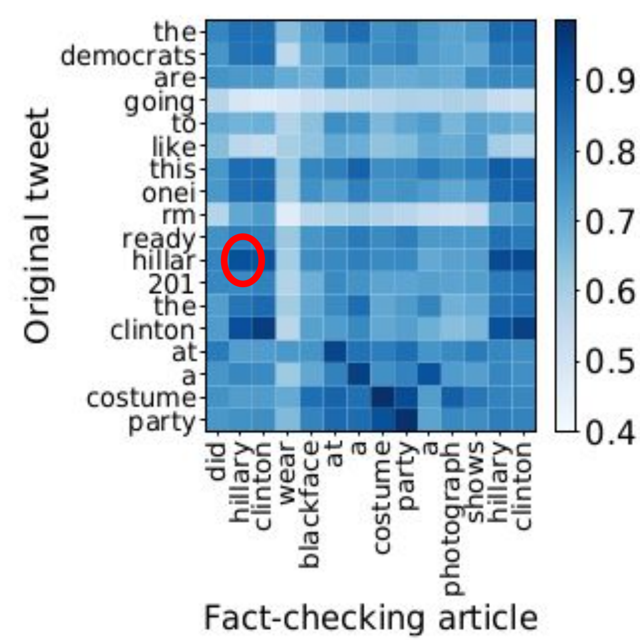
(a) Matrix $\mathbf{S}$ in Eq. 3

ELMo Embedding Interactions

Contextual Embeddings able to capture high similarity between a typo and a normal word

$$\mathbf{C}_{ij} = \frac{{\mathbf{h}_i^q}^T \cdot \mathbf{h}_j^d}{||\mathbf{h}_i^q|| \times ||\mathbf{h}_j^d||}, \mathrm{i} = 1..\mathrm{N}, \mathrm{j} = 1..\mathrm{M}$$

Recall that $\mathbf{h}$ is the output of the Projection Layer for ELMo Embedding

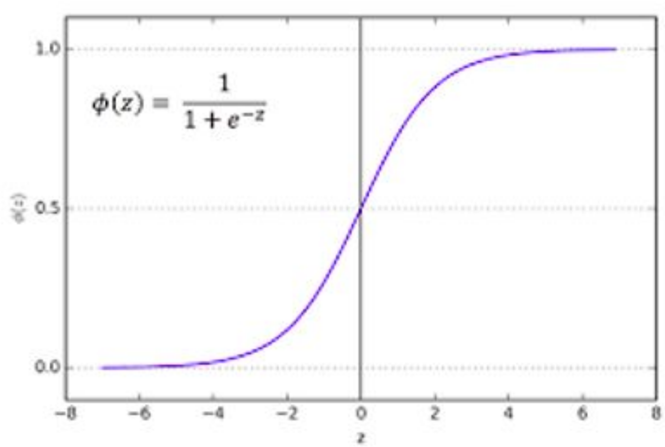


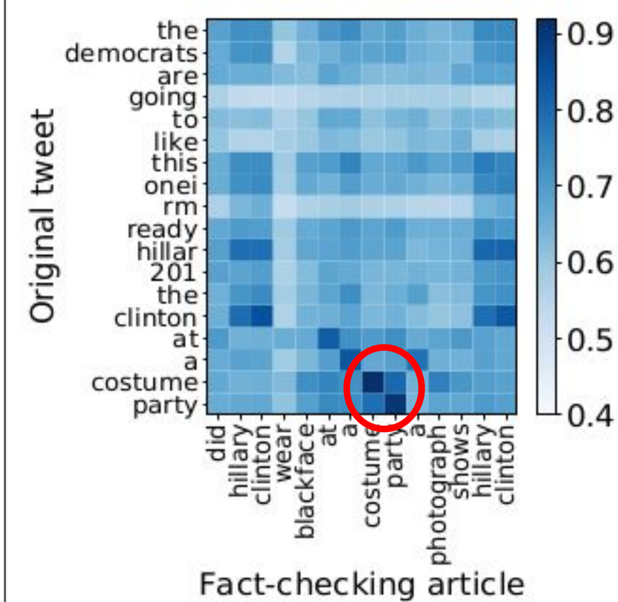
(d) Matrix **C** in Eq. 6

Attended Interaction Matrix

Attention mechanism to avoid over-reliance of raw similarities from projected GloVe Embeddings

$$\mathbf{G}_{ij} = 2 \times \boldsymbol{\sigma}(-||\mathbf{h}_i^q - \mathbf{h}_j^d||), \mathrm{i} = 1..\mathrm{N}, \mathrm{j} = 1..\mathrm{M}$$

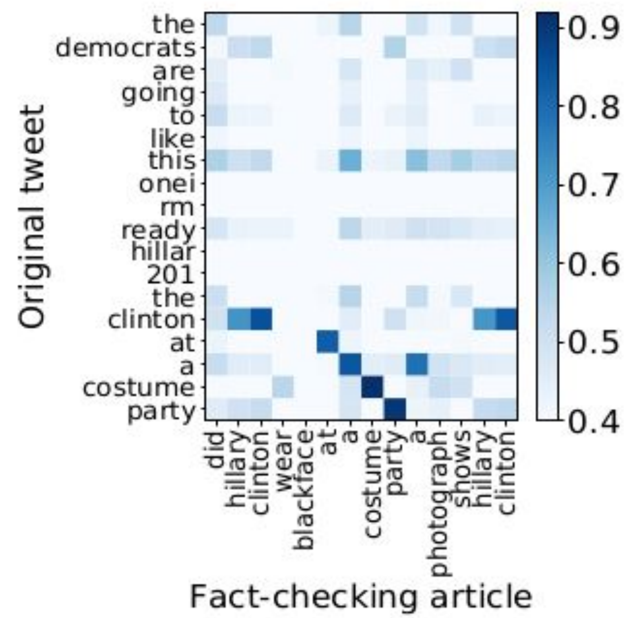




(b) Matrix **G** in Eq. 4

Attended Interaction Matrix

$$\mathbf{A}_{ij} = \mathbf{S}_{ij} \times \mathbf{G}_{ij}, \mathrm{i} = 1..\mathrm{N}, \mathrm{j} = 1..\mathrm{M}$$



(c) Matrix **A** in Eq. 5

Intuition (Open to Discussion)

S - Similarity between GloVe embeddings

G - Similarity between ELMo embeddings

A - Extent to which G attends to S

W1 - word in query (tweet), **W2** - word in document (FC-article)

a) If W1, W2 are different, and occur in same context

S=0, G=1, A=0

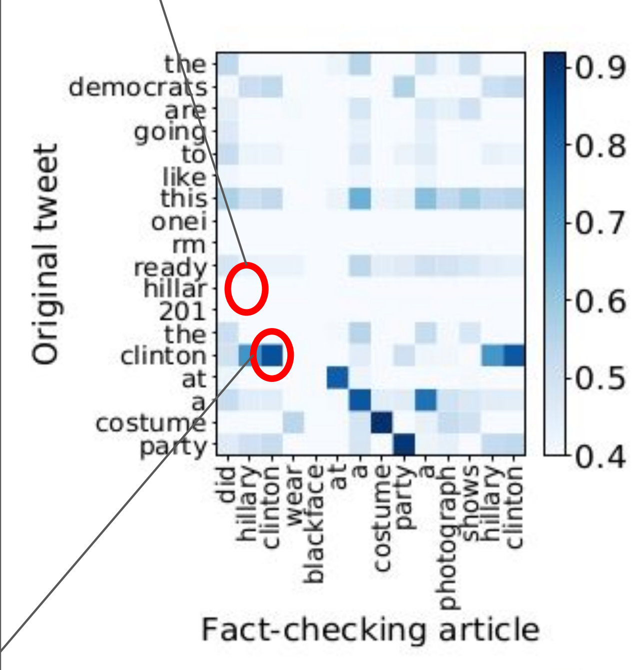
b) If W1, W2 are same, but occur in differ contexts(different word sense)

S=1, G=0, A=0

c) If W1, W2 are same, occur in same context

S=1, G=1, A=1

hillar, hillary
case (a)



(c) Matrix **A** in Eq. 5

clinton, clinton
case (c)

Textual Feature Extraction

Matrix S - Glove Embedding Interaction

Matrix A - Attended Interaction Matrix

Matrix C - ELMo Embedding Interaction

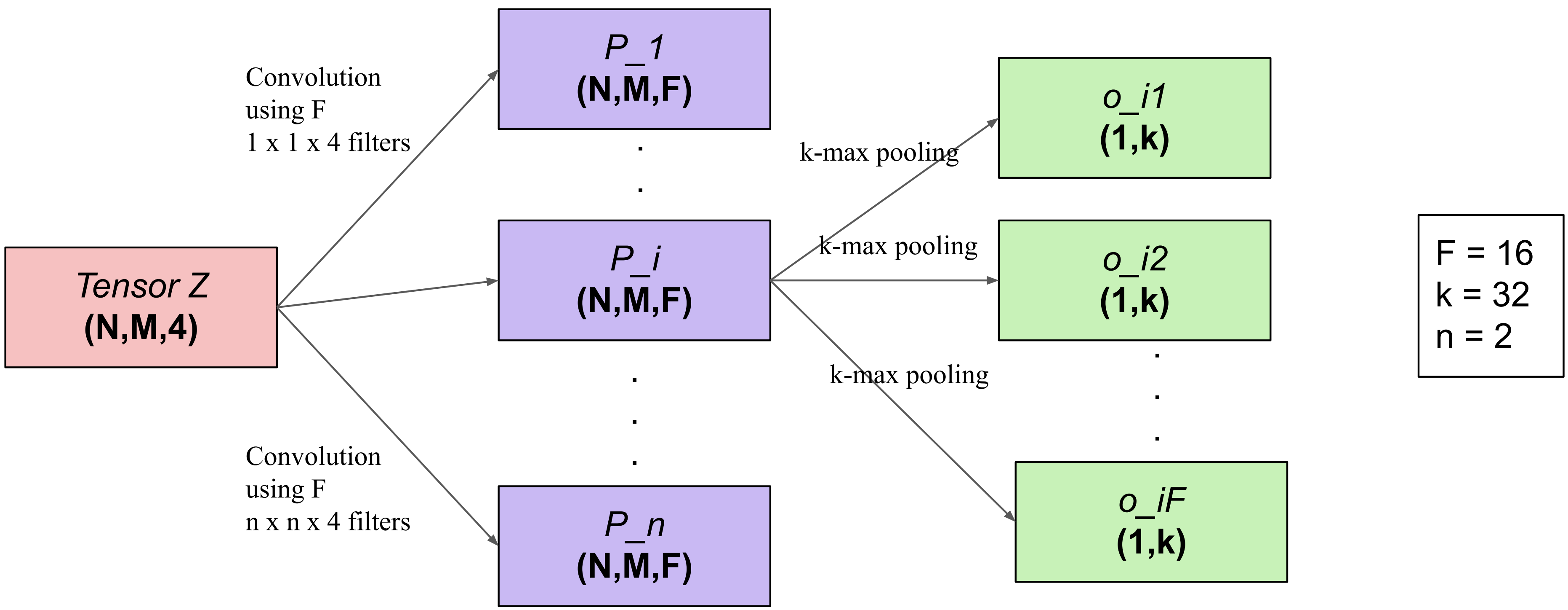
Matrix (S - C) - To make the model aware of difference between interaction matrices

---

Stack all the 4 matrices, each of dimension (N,M) to get a 3-D tensor ***Z*** *of dimension* ***(N,M,4)***

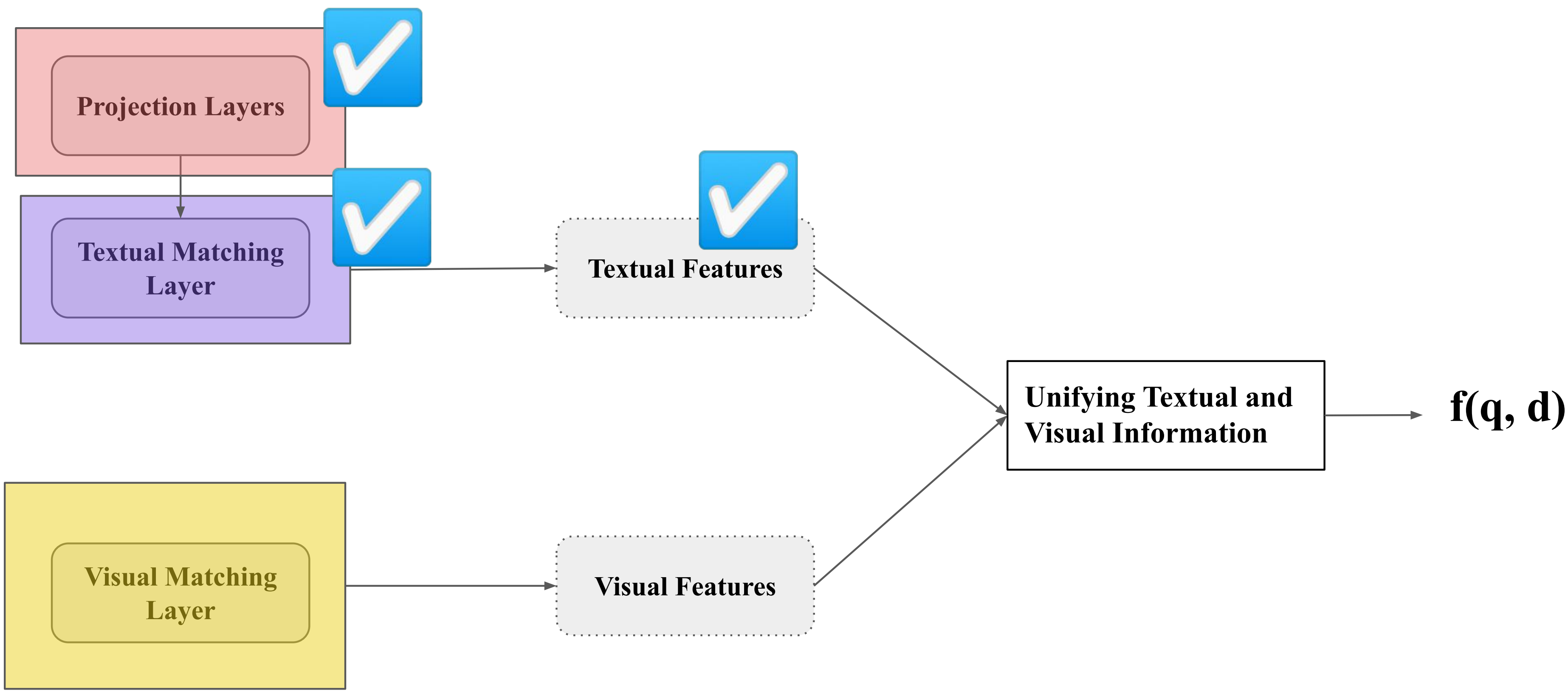
$$\mathbf{Z} = [\mathbf{S} \oplus \mathbf{A} \oplus \mathbf{C} \oplus (\mathbf{S} - \mathbf{C})]$$

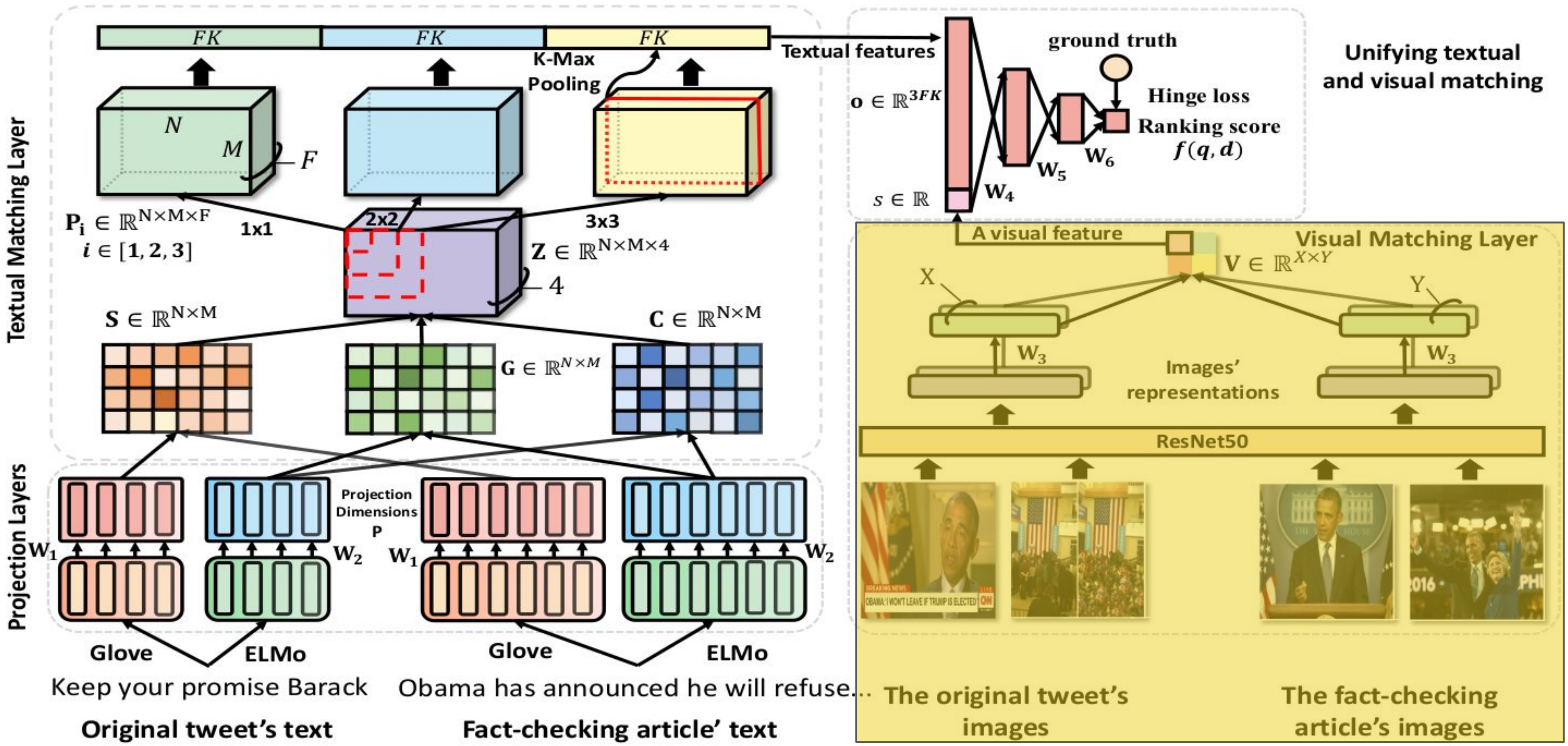
Textual Feature Extraction



o = [o_11; o_12; ….o_1F;........;o_i1;....o_iF;.......;o_n1;.....;o_nF] — *Dimension of o?*

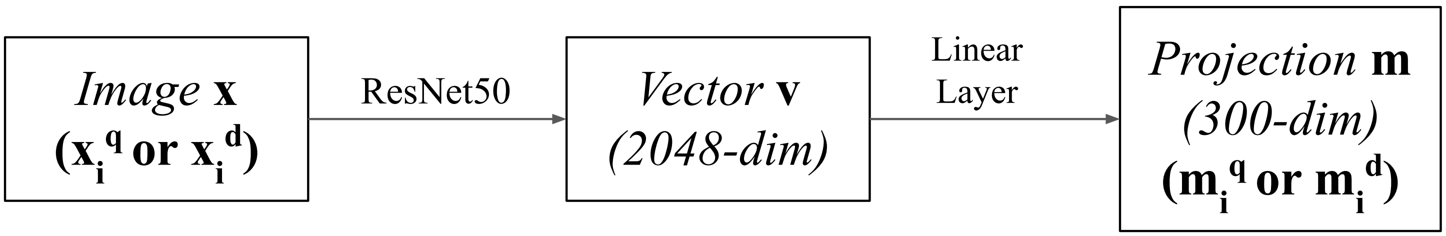
Framework: Multimodal Attention Network - MAN





Visual Matching Layer

Visual Matching Layer



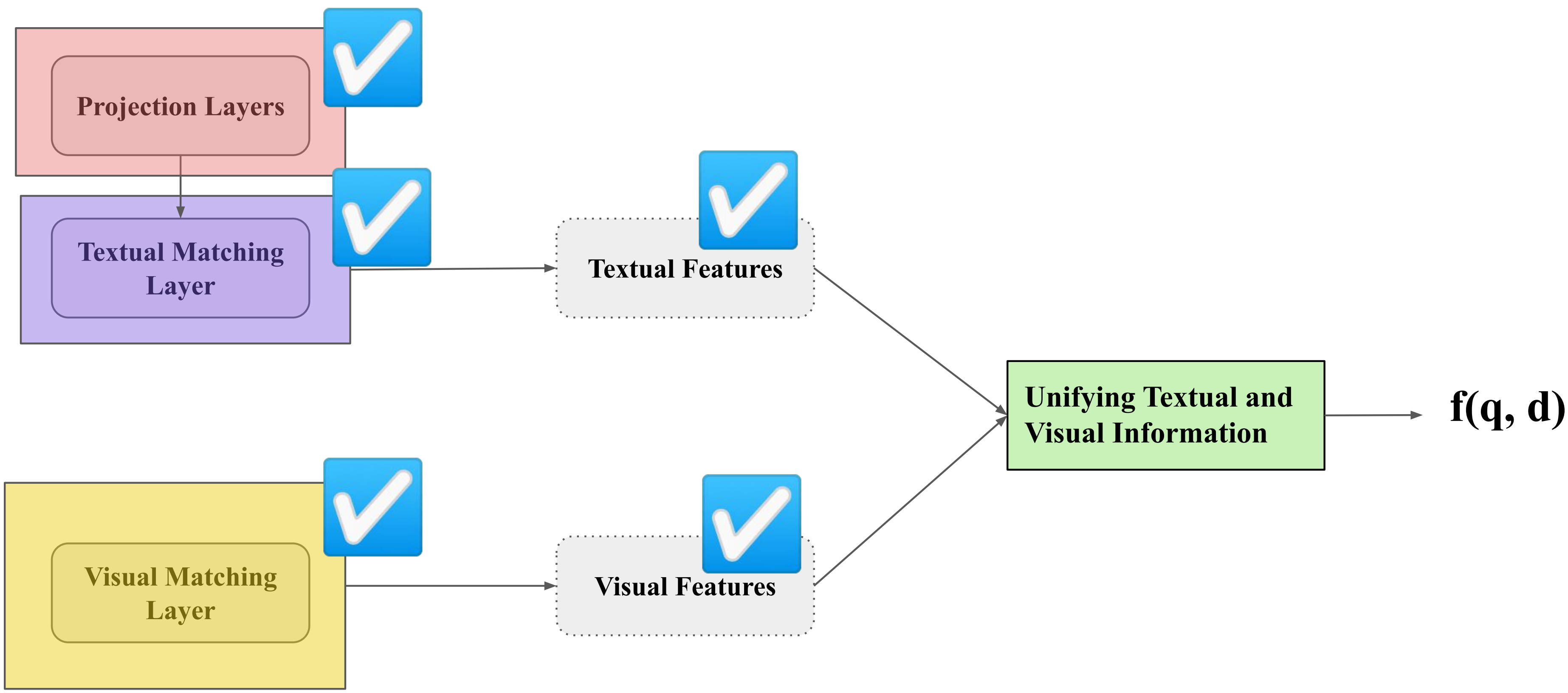
Document (FC-article) is relevant to a query (tweet) if they have similar images

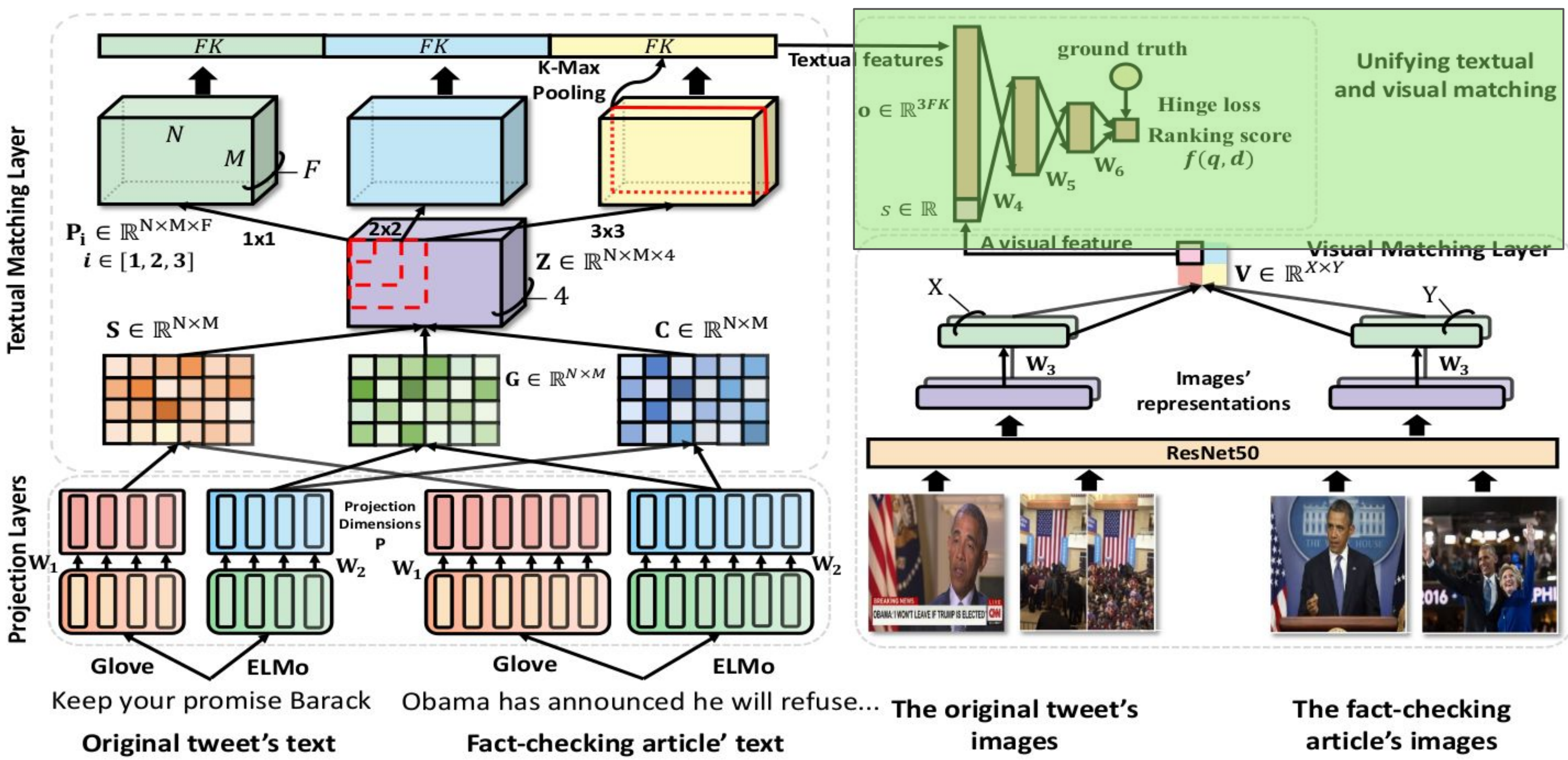
$$\mathbf{V}_{ij} = \frac{\mathbf{m}_i^{q^T} \cdot \mathbf{m}_j^d}{||\mathbf{m}_i^q|| \times ||\mathbf{m}_j^d||}, \mathrm{i} = 1..\mathrm{X}, \mathrm{j} = 1..\mathrm{Y}$$

$s = max(\mathbf{V})$, where $s \in \mathbb{R}$

If article has no images,
s = -1

Framework: Multimodal Attention Network - MAN





Textual Matching Layer
Projection Layers
FK
FK
FK
K-Max
Pooling
N
M
F
$\mathbf{P_i} \in \mathbb{R}^{N\times M\times F}$
$i \in [1,2,3]$
1x1
2x2
3x3
$\mathbf{Z} \in \mathbb{R}^{N\times M\times 4}$
4
$\mathbf{S} \in \mathbb{R}^{N\times M}$
$\mathbf{G} \in \mathbb{R}^{N\times M}$
$\mathbf{C} \in \mathbb{R}^{N\times M}$
Projection
Dimensions
P
$\mathbf{W_1}$
$\mathbf{W_2}$
$\mathbf{W_1}$
$\mathbf{W_2}$
Glove
ELMo
Glove
ELMo
Keep your promise Barack
Obama has announced he will refuse...
Original tweet's text
Fact-checking article' text
Textual features
$\mathbf{o} \in \mathbb{R}^{3FK}$
$s \in \mathbb{R}$
ground truth
Hinge loss
Ranking score
$f(q,d)$
$\mathbf{W_4}$
$\mathbf{W_5}$
$\mathbf{W_6}$
Unifying textual
and visual matching
A visual feature
Visual Matching Layer
$\mathbf{V} \in \mathbb{R}^{X\times Y}$
X
Y
$\mathbf{W_3}$
$\mathbf{W_3}$
Images'
representations
ResNet50
The original tweet's
images
The fact-checking
article's images

Unifying Textual and Visual Information

Textual Features: **o** = [o_11; o_12; ….o_1F;........;o_i1;....o_iF;.......;o_n1;.....;o_nF]

Visual Features: **s** = max(V)

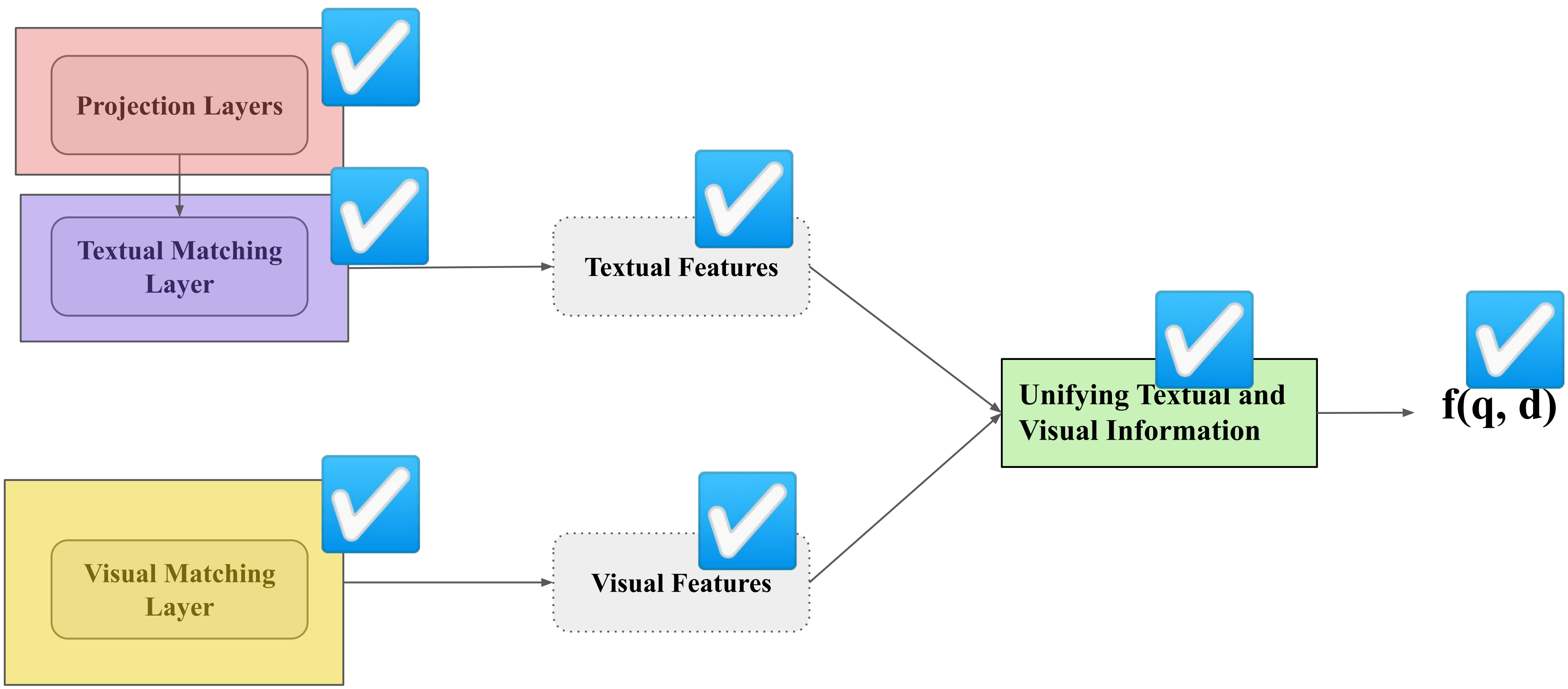
Input features: **[o ; s]** - concatenate **o** and **s** *(Dimension:* ***nfk + 1)***

$$f(q,d) = \mathbf{W}_6 \cdot relu(\mathbf{W}_5 \cdot relu(\mathbf{W}_4 \cdot \underbrace{[\mathbf{o}; s]}_{\textit{Input Feature}}))$$

<u>Minimize Hinge Loss</u>

$$\mathcal{L}(q,d^+,d^-) = max(0, 1 - f(q,d^+) + f(q,d^-))$$

Framework: Multimodal Attention Network - MAN



Data Collection

- Checking FC-articles is laborious
- Pairs of *(**Original Tweet , FC-articles** - embedded in replies to original tweet)*[1]
- *FC-articles from 2 major sites:* ***snopes.com***[2], ***politifact.com***[3]
- From the original tweet replies, pairs of tweet **q** and FC-article **d** are generated - ***(q, d)***

Only tweets with both text and images are kept

19341 pairs of (Tweet , FC-article)

Manual fact-checking done (label 1 or 0 depending on whether ***d fact-checks q -*** *Majority voting by 3 labelers)*

[1] Nguyen Vo and Kyumin Lee. 2019. Learning from fact-checkers: Analysis and generation of fact-checking language. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 335–344.
[2] **https://www.snopes.com/**
[3] **https://www.politifact.com/**

Data Collection

Moderate agreement between 3 labelers

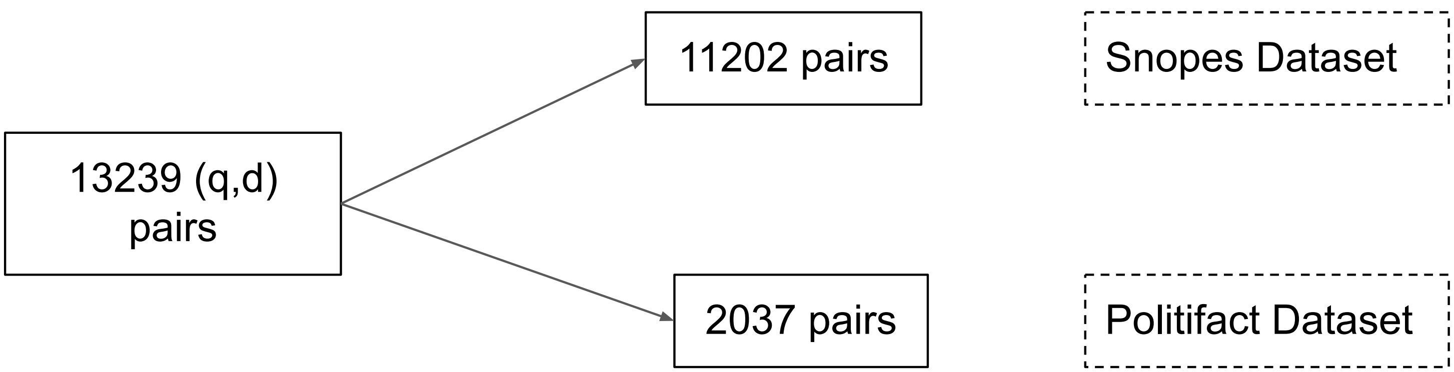
Reason: FC-article and tweet are topically similar but article **does not** fact-check the tweet

Many tweets related to Donald Trump and Hillary Clinton - collected during 2016 US Presidential Election

19341 pairs of (Tweet , FC-article) - reduced to ***13239 positive pairs***

Lot of ***False Negatives*** *- Because some FC-articles may not have been included in the reply to the tweet.*

Data Collection



- **102 Overlapping tweets**
- **False Negatives still exist, but the number is smaller than that of the full dataset**

Evaluation Metric

1. Normalized Discounted Cumulative Gain (**NDCG@K**)
2. **HIT@K**

Normalized Discounted Cumulative Gain (NDCG)

Consider 3 Ranking systems: a) **System A**, b) **System B**, c) **Ideal System**

We have a query **q,** and 3 documents **d1, d2, d3**

Possible relevance scores: {**0** - irrelevant, **1** - moderately relevant, **2** - very relevant}

For query q, relevance scores for the 3 documents are :

relevance(**q**, **d1**) = **0**

relevance(**q**, **d2**) = **1**

relevance(**q**, **d3**) = **2**

For an **Ideal ranking system**, what is the correct order of ranking of documents, given query **q**?

Normalized Discounted Cumulative Gain (NDCG)

Consider 3 Ranking systems: a) **System A**, b) **System B**, c) **Ideal System**

We have a query **q,** and 3 documents **d1, d2, d3**

Possible relevance scores: {**0** - irrelevant, **1** - moderately relevant, **2** - very relevant}

For query q, relevance scores for the 3 documents are :

relevance(**q**, d1) = **0**

relevance(**q**, d2) = **1**

relevance(**q**, d3) = **2**

For an **Ideal ranking system**, what is the correct order of ranking of documents, given query **q**?

d3 d2 d1

Normalized Discounted Cumulative Gain (NDCG)

Rank	System A	System B	Ideal System
1	**d2 (1)**	**d1 (0)**	**d3 (2)**
2	**d3 (2)**	**d2 (1)**	**d2 (1)**
3	**d1 (0)**	**d3 (2)**	**d1 (0)**

Normalized Discounted Cumulative Gain (NDCG)

Rank	System A	System B	Ideal System
1	**d2 (1)**	**d1 (0)**	**d3 (2)**
2	**d3 (2)**	**d2 (1)**	**d2 (1)**
3	**d1 (0)**	**d3 (2)**	**d1 (0)**

Cumulative Gain:

System A = 1 + 2 + 0 = **3**

System B = 0 + 1 + 2 = **3**

Ideal System = 2 + 1 + 0 = **3**

Normalized Discounted Cumulative Gain (NDCG)

Rank	System A	System B	Ideal System
1	d2 (1)	d1 (0)	d3 (2)
2	d3 (2)	d2 (1)	d2 (1)
3	d1 (0)	d3 (2)	d1 (0)

Cumulative Gain:

System A = 1 + 2 + 0 = **3**

System B = 0 + 1 + 2 = **3**

Ideal System = 2 + 1 + 0 = **3**

$$DCG = \sum_{i=1}^{n} \frac{2^{relevance_i} - 1}{log_2(i+1)}$$

Discounted Cumulative Gain:

$$System A : \frac{2^1 - 1}{\log_2 2} + \frac{2^2 - 1}{\log_2 3} + \frac{2^0 - 1}{\log_2 4} = 2.9$$

$$System B : \frac{2^0 - 1}{\log_2 2} + \frac{2^1 - 1}{\log_2 3} + \frac{2^2 - 1}{\log_2 4} = 2.13$$

$$Ideal\ System : \frac{2^2 - 1}{\log_2 2} + \frac{2^1 - 1}{\log_2 3} + \frac{2^0 - 1}{\log_2 4} = 3.63$$

Normalized Discounted Cumulative Gain (NDCG)

Rank	System A	System B	Ideal System
1	d2 (1)	d1 (0)	d3 (2)
2	d3 (2)	d2 (1)	d2 (1)
3	d1 (0)	d3 (2)	d1 (0)

Cumulative Gain:

System A = 1 + 2 + 0 = **3**

System B = 0 + 1 + 2 = **3**

Ideal System = 2 + 1 + 0 = **3**

$$DCG = \sum_{i=1}^{n} \frac{2^{relevance_i} - 1}{log_2(i+1)}$$

Discounted Cumulative Gain:

$$System A: \frac{2^1-1}{\log_2 2} + \frac{2^2-1}{\log_2 3} + \frac{2^0-1}{\log_2 4} = 2.9$$

$$System B: \frac{2^0-1}{\log_2 2} + \frac{2^1-1}{\log_2 3} + \frac{2^2-1}{\log_2 4} = 2.13$$

$$Ideal\ System: \frac{2^2-1}{\log_2 2} + \frac{2^1-1}{\log_2 3} + \frac{2^0-1}{\log_2 4} = 3.63$$

Normalized DCG - NDCG:

System A: (2.9/3.63) = **0.80**
System B: (2.13/3.63) = **0.59**

Normalized Discounted Cumulative Gain (NDCG)

Rank	System A	System B	Ideal System
1	d2 (1)	d1 (0)	d3 (2)
2	d3 (2)	d2 (1)	d2 (1)
3	d1 (0)	d3 (2)	d1 (0)

Cumulative Gain:

System A = 1 + 2 + 0 = **3**

System B = 0 + 1 + 2 = **3**

Ideal System = 2 + 1 + 0 = **3**

$$DCG = \sum_{i=1}^{n} \frac{2^{relevance_i} - 1}{log_2(i+1)}$$

Discounted Cumulative Gain:

$$System A : \frac{2^1 - 1}{\log_2 2} + \frac{2^2 - 1}{\log_2 3} + \frac{2^0 - 1}{\log_2 4} = 2.9$$

$$System B : \frac{2^0 - 1}{\log_2 2} + \frac{2^1 - 1}{\log_2 3} + \frac{2^2 - 1}{\log_2 4} = 2.13$$

$$Ideal\ System : \frac{2^2 - 1}{\log_2 2} + \frac{2^1 - 1}{\log_2 3} + \frac{2^0 - 1}{\log_2 4} = 3.63$$

Normalized DCG - NDCG:

NDCG@3

System A: (2.9/3.63) = **0.80**
System B: (2.13/3.63) = **0.59**

HIT@K

Consider a tweet (query) **q**

There are 10 documents **d1, d2,......., d10**

Let the **true (most relevant)** FC-articles associated with it be : **d1, d4, d9, d10**

HIT@K = Fraction of true articles in the top K ranking predictions

System C ranking: **d6, d1, d3, d5, d9, d4, d8, d10, d7, d2**

HIT@1 =
0/4 = 0.00

HIT@3 =
¼ = 0.25

HIT@7 =
¾ = 0.75

BM25 (Best Match 25)

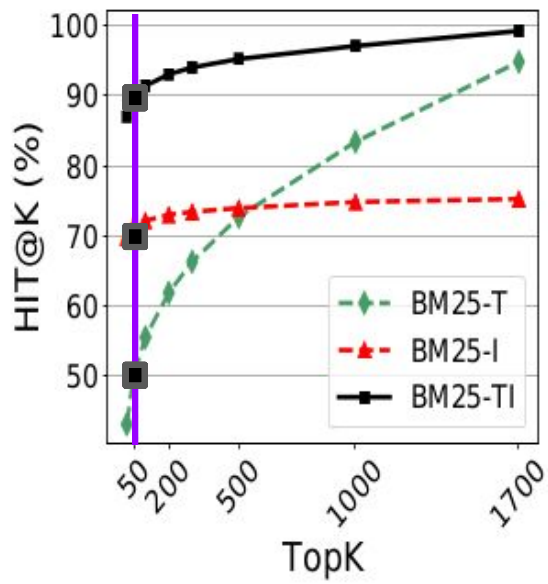
BM25 is a **ranking function** that ranks **documents** that are relevant to a given **query** in the decreasing order of relevance (most relevant to least relevant)

BM25-T: queries are tweets’ text

BM25-I: queries are text in tweets’ images

BM25-TI: queries are tweets’ text + text in tweets’ images

BM25



(a) Snopes (b) Politifact

Figure 3: Performance of basic retrieval methods

In Fig (a),

HIT@K (K=50) for
BM25-T : **50%**
BM25-I : **70%**

Suggests lot of fake news appears in images. Images are more attractive to online users, and easier to convey fake news. Moreover, tweets' texts are less than 280 characters

BM25-I saturates quickly as K increases. Only some queries have text in images

Best performance for BM25-TI, and hence used.

Split Dataset

What is a good value for K? *[K - number of initial candidates - output of BM25-TI]*

If K is too small, there may not be relevant articles associated with the candidates

If K is too large, reranking system takes a lot of time to run

So, K = 50

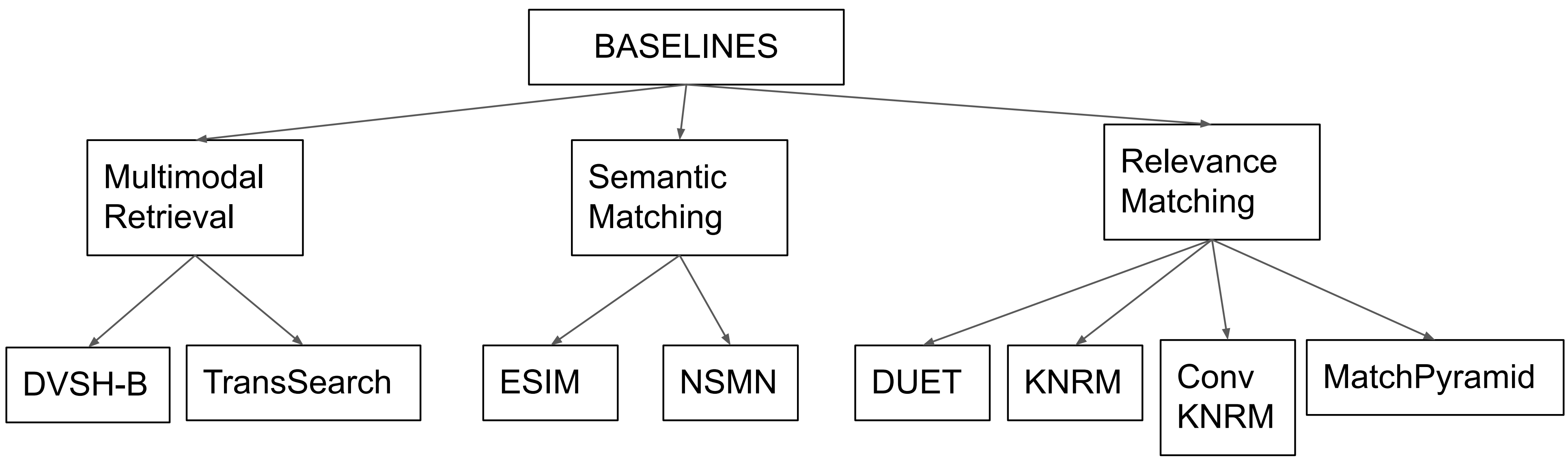
Table 1: Split datasets

Datasets	Snopes			Politifact		
Items	Train	Valid	Test	Train	Valid	Test
|Original Tweets|	8,002	1,000	1,001	1,496	187	187
|FC-Articles|	1,703	1,697	1,697	467	467	467

Testing Scenario

SC1 - text and images from both tweets and FC-articles

SC2 - text and images from both tweets and FC articles + text from images



Testing Scenario 1 - SC1

Table 2: Performance of our models and baselines when using images and text in tweets

Ranking Models Types	Ranking Models	Snopes					Politifact				
		NDCG@1	NDCG@3	HIT@3	NDCG@5	HIT@5	NDCG@1	NDCG@3	HIT@3	NDCG@5	HIT@5
Exact Matching	BM25-T	0.20579	0.27642	0.32867	0.30420	0.39461	0.18182	0.29162	0.37968	0.31348	0.43316
Multimodal Retrieval (Group 1)	DVSH-B	0.38661	0.51091	0.60040	0.54084	0.67333	0.26203	0.33333	0.38503	0.36003	0.44920
	TransSearch	0.31668	0.46081	0.56444	0.50062	0.66034	0.28342	0.37925	0.44920	0.40040	0.50267
Semantic Matching (Group 2)	ESIM	0.33367	0.46608	0.56444	0.50372	0.65534	0.14973	0.28722	0.39037	0.34871	0.53476
	NSMN	0.45754	0.60097	0.70330	0.63220	0.77822	0.37968	0.47718	0.55080	0.53128	0.67914
Relevance Matching (Group 3)	DUET	0.36863	0.48875	0.57842	0.52628	0.66833	0.29412	0.41009	0.49733	0.43505	0.55615
	MatchPyramid	0.48052	0.58523	0.66034	0.61565	0.73327	0.29412	0.38903	0.45455	0.40812	0.50267
	KNRM	0.48951	0.61081	0.69730	0.63686	0.76124	0.42246	0.54935	0.63636	0.58456	0.72193
	ConvKNRM	0.52148	0.63168	0.70929	0.65942	0.77522	0.45989	0.57229	0.65241	0.62117	0.77005
	CoPACRR	0.53247	0.64469	0.72328	0.67208	0.78921	0.45455	0.59344	0.69519	0.62761	0.77540
Ours	CTM	0.55744	0.67555	0.75624	0.70156	0.81918	0.47059	0.61669	0.71658	0.64292	0.78075
	VMN	0.68931	0.73540	0.76723	0.75019	0.80320	0.24599	0.26821	0.31551	0.28363	0.35829
	MAN	**0.74326**	**0.82197**	**0.87712**	**0.83447**	**0.90609**	**0.55080**	**0.65435**	**0.73262**	**0.67644**	**0.78610**
MAN vs. the best result of baselines		39.59%	27.50%	21.27%	24.16%	14.81%	19.77%	10.26%	5.38%	7.78%	1.38%

CTM outperforms the best baselines; VMN outperforms text-based ranking baselines in Snopes

Testing Scenario 2 - SC2

Table 3: Performance of our models and baselines when using images, text in tweets and text in images

Ranking Models Types	Ranking Models	Snopes					Politifact				
		NDCG@1	NDCG@3	HIT@3	NDCG@5	HIT@5	NDCG@1	NDCG@3	HIT@3	NDCG@5	HIT@5
Exact Matching	BM25-TI	0.63736	0.69650	0.73826	0.71058	0.77223	0.27807	0.34928	0.40642	0.38909	0.50267
Multimodal Retrieval (Group 1)	DVSH-B	0.32667	0.46849	0.56843	0.49640	0.63636	0.21925	0.29335	0.34759	0.32626	0.42246
	TransSearch	0.45854	0.58410	0.67433	0.61832	0.75724	0.39572	0.50878	0.58824	0.52397	0.62567
Semantic Matching (Group 2)	ESIM	0.61139	0.70660	0.77323	0.72999	0.83117	0.33155	0.44658	0.52941	0.48617	0.62567
	NSMN	0.78821	0.85732	0.90809	0.87148	0.94106	0.58824	0.70002	0.77540	0.73500	0.86096
Relevance Matching (Group 3)	DUET	0.51848	0.63605	0.71928	0.67075	0.80220	0.41711	0.53087	0.60963	0.55757	0.67380
	MatchPyramid	0.86513	0.91150	0.94406	0.91791	0.95904	0.64171	0.74872	0.82353	0.77702	0.89305
	KNRM	0.84815	0.89118	0.92008	0.90271	0.94805	0.65775	0.75464	0.82353	0.77237	0.86631
	ConvKNRM	0.85914	0.90829	0.94306	0.91401	0.95704	0.66310	0.79163	0.88235	0.80705	0.91979
	CoPACRR	0.86913	0.91166	0.94006	0.91851	0.95604	0.66845	0.77419	0.84492	0.79191	0.88770
Ours	CTM	0.89910	0.93191	0.95504	0.94008	0.97502	0.71123	0.82512	0.89840	0.84331	0.94118
	MAN	0.88412	0.92563	0.95604	0.93238	0.97203	0.72193	0.83104	0.90374	0.85313	**0.95722**
	MAN-A	**0.90909**	**0.94204**	**0.96503**	**0.94892**	**0.98202**	**0.74332**	**0.84905**	**0.91979**	**0.85987**	0.94652
MAN-A vs. best result of baselines		4.60%	3.33%	2.22%	3.31%	2.40%	11.20%	7.25%	4.24%	6.54%	2.91%

Conclusion

- Authors present a novel framework to alleviate the spread of fake news and increase verified content on social media
- Authors compare their approach with a variety of ranking functions
- The framework MAN, using textual and visual information, outperforms all the ranking baseline methods on NDCG@K and HIT@K.
- Very well curated dataset
- Authors don't directly address the question "How do we reduce the spread of fake news". I believe it's more of a consequence of developing a good ranking function and retrieving correct FC-articles.

THANK YOU